Poznań University of Technology

Faculty of Automatic Control, Robotics and Electrical Engineering

Institute of Automatic Control and Robotics

Division of Electronic Systems and Signal Processing

Highly efficient night-vision pedestrian detection based on thermal images

Ph.D. Dissertation

Karol Piniarski

Supervisor: Prof. dr hab. eng. Adam Dąbrowski
Auxiliary supervisor: Dr eng. Paweł Pawłowski

Poznań 2021

Acknowledgements

First I would like to thank my supervisor
prof. dr hab. eng. Adam Dąbrowski
for his invaluable advice
and continuous support
during my Ph.D. study.

I would like to thank my auxiliary supervisor
dr eng. Paweł Pawłowski
for his constant support,
hard work and a lot of patience
while offering his advice.

I would like also to express
gratitude to my wife
for her understanding and support
in the past few years.

Contents

List of abbreviations

ACF – aggregated channel feature
AdaBoost – adaptive boosting
ADAS – advanced driver assistance systems
ADT – adaptive dual-threshold
CCTV – closed circuit television
CNN – convolutional neural network
CPU – central processing unit
CVC – Computer Vision Center
FIR – far infrared
FPPI – false-positives per image
FPS – frames per second
GPGPU – general-purpose computing on graphics processing units
HOG – histograms of oriented gradients
ICF – integral channel features
IR – infrared
ITS – intelligent transportation systems
KAIST – Korea Advanced Institute of Science and Technology
LAMR – log-average miss rate
LED – light-emitting diode
LSI – Laboratorio de Sistemas Inteligentes
LWIR – long-wave-infrared
MCT – mean calculation time
MR – miss rate
NIR – near-infrared
NTPD – night-time pedestrian dataset
OSU – Ohio State University
PR – the number of selected ROIs per frame
R-CNN – region-based convolutional neural network
RGB – reed green blue
ROI – region of interest
RPN – region proposal networks
SVM – support-vector machine
TPU – tensor processing unit
YOLO – you only look once

List of symbols

W_{tot} – total radiation power
ε – emissivity of the object
τ – transmission through the atmosphere
T_{obj} – object temperature
T_{atm} – atmosphere temperature
T_{Otsu} – Otsu threshold
α_{caf} – constant adjusting factor
β – difference factor
T_{L} – lower threshold
T_{M} – medium threshold
T_{H} – higher threshold
A_{obj1}, $\mathrm{A}_{\mathrm{obj2}}$ – areas of the objects (in pixels)
$A_{\mathrm{obj1}\cap\mathrm{obj2}}$ – area of the intersection of objects
α_{sim} – similarity coefficient
α_{h} – height coefficient
μ – mean value of all pixels in ROI
α_{σ} – homogenous coefficient
η_{20}, η_{02} – second-order normalized central moments
μ_{pq} – central moment
α_{η} – skew threshold for normalized central moments
α_{f} – threshold for fill factor
$\alpha_{\mathrm{HW}_{\mathrm{min}}}$ – minimum height-to-width ratio
$\alpha_{\mathrm{HW}_{\mathrm{max}}}$ – maximum height-to-width ratio
A_{init} – minimum object area
A_{ROI} – minimum ROI area
$l_{\mathrm{ROIs}_{\mathrm{max}}}$ – maximum number of ROIs per one image
k – scale factor
ρ – performance index
w_{ρ} – performance index weight
a – overall accuracy
$\mathrm{FPS}_{\mathrm{max}}$ – maximum value of FPS achieved over all possible resolutions
$\mathrm{FPS}_{\mathrm{cal}}$ – calculated value of FPS with a given resolution
t_{cal} – mean calculation time of one test sample with a given resolution
t_{min} – minimum calculation time achieved over all possible resolutions

Abstract

The scientific aim of this Ph.D. dissertation is the analysis and development of automated mechanism for highly efficient night-vision pedestrian detection on thermal images. The research presented in this dissertation is focused to two main issues: region of interest (ROI) generation based on thresholding and procedure of tuning object classification stage. The author's motivation was to achieve the state-of-the-art accuracy and real-time performance of pedestrian detection process in order to apply it in vehicles (e.g. those equipped with driver assistance systems or in autonomous vehicles) without using special hardware i.e., general-purpose computing on graphics processing units.

The scientific thesis was formulated as follows: The developed approach of night-vision pedestrian detection based on proposed ROI generation by thresholding of thermal images and by properly tuned object classification procedure improves detection accuracy and significantly increases computational efficiency of the pedestrian detection process.

The structure of this dissertation is as follows: after the introduction, in Chapter 2, the extended analysis of the research area (initially described in Chapter 1.1) is presented along with a summary and detailed explanation of the motivation to undertake this work.

Section 2.2 presents all the public recording datasets and benchmarks used in the experiments. Subsequently, the proposed improvements to the pedestrian detection process are presented in separated chapters.

The proposed ROI generation algorithm is described and tested in Chapter 3. In the next Chapter 4, the problem of inaccurate matching of the edges of ROI to the outer edges of the pedestrians is analysed with the proposed additional ROI area enlarging technique. Chapter 5 describes the proposed procedure of tuning object classification stage with universal performance index.

Chapter 6 presents the experiments on the proposed pedestrian detection algorithm performed to compare the presented solution with the standard approaches based on the sliding window segmentation technique and other solutions in the literature. Chapters 3 and 5 also include separated experiments and tests performed to verify the effectiveness of the proposed modifications.

In the Chapter 7, the author also presents research on the possibility of using so-called multi-spectral vision for scene analysis by monitoring operators. Performed experiments show that this option shortens reactions and supports faster identification of objects (e.g., pedestrians) at night.

The last chapter presents the conclusions, which indicate that the scientific goal of this dissertation has been achieved and the scientific thesis has been proven. The author’s approach to night-vision pedestrian detection achieved very high computational efficiency, with up to 130 frames per second using the CPU only. Moreover, it was possible to obtain the state-of-the-art detection accuracy for tested detectors, namely the aggregated channel feature (ACF) and deep convolutional neural network (CNN).

Streszczenie

Celem naukowym prezentowanej rozprawy doktorskiej jest analiza i opracowanie automatycznego mechanizmu detekcji pieszych na obrazach termowizyjnych rejestrowanych w nocy. Badania przedstawione w niniejszej rozprawie koncentrują się na dwóch głównych zagadnieniach: ekstrakcji obszaru zainteresowania w oparciu o progowanie obrazu termowizyjnego oraz odpowiednim dopasowaniu procedury klasyfikacji obiektów. Motywacją autora było osiągnięcie wysokiej wydajności procesu detekcji pieszych przy jednoczesnym zachowaniu wysokiej dokładności. Głównym zastosowaniem systemu jest detekcja w pojazdach (w systemach czasu rzeczywistego) bez konieczności wykorzystania dedykowanego sprzętu, tj. procesory graficzne.

Została sformułowana następująca teza naukowa pracy: Opracowane podejście do detekcji pieszych w nocy w oparciu o zaproponowany proces ekstrakcji obszaru zainteresowania poprzez progowanie obrazów termowizyjnych oraz odpowiednio dostrojoną procedurę klasyfikacji obiektów poprawia dokładność detekcji i znacząco zwiększa wydajność obliczeniową.

Struktura pracy jest następująca: po wstępie w Rozdziale 2. przedstawiona jest rozszerzona analiza obszaru badawczego (opisana wstępnie w Rozdziale 1.1.) wraz z podsumowaniem i szczegółowym wyjaśnieniem motywacji do podjęcia tej pracy.

Sekcja 2.2. przedstawia wszystkie wykorzystane w eksperymentach publiczne zbiory danych. Następnie w osobnych rozdziałach przedstawiono proponowane usprawnienia procesu detekcji pieszych.

Proponowany algorytm ekstrakcji obszaru zainteresowania został przedstawiony i przetestowany w Rozdziale 3. W następującym Rozdziale 4. przeanalizowano problem niedokładnego dopasowania wyodrębnionego obszaru do zewnętrznych krawędzi pieszego wraz z proponowaną techniką jego powiększania. W rozdziale 5 przedstawiono i opisano proponowaną procedurę dostrajania etapu klasyfikacji obiektów z zaproponowanym przez autora uniwersalnym indeksem wydajności.

W rozdziale 6. przedstawiono eksperymenty z proponowanym algorytmem detekcji pieszych uwzględniającym wprowadzone ulepszenia. Przedstawiono porównanie proponowanego podejścia ze standardowym algorytmem detekcji opartymi na technice przesuwnego okna oraz innymi rozwiązaniami prezentowanymi w literaturze. Rozdziały 3. i 5. zawierają również oddzielne eksperymenty i testy przeprowadzone w celu sprawdzenia skuteczności poszczególnych modyfikacji.

W Rozdziale 7. autor przedstawia badania nad możliwością wykorzystania przez operatorów monitoringu tzw. obrazowania wielospektralnego. Przeprowadzone eksperymenty pokazują, że proponowane rozwiązanie skraca reakcję i wspomaga manualną identyfikację pieszych w nocy.

W ostatnim rozdziale zawarto podsumowanie, które dowodzi że cel naukowy niniejszej rozprawy został zrealizowany, a teza naukowa udowodniona. Proponowane przez autora podejście do detekcji pieszych w nocy osiągnęło bardzo wysoką wydajność obliczeniową przy użyciu samego procesora. Ponadto udało się uzyskać wysoką dokładność detekcji dla testowanych detektorów.

1. Introduction

1.1. Research area

The dynamic growth of motorization and the increased traffic volume, although together help to develop our civilization, also increase the risk of accidents. According to [1], 38% of fatal accidents in the European Union occur in darkness, despite the fact that the traffic during nights is several times smaller than on days. About 20% of the victims are pedestrians, while more than half of pedestrian deaths (51%) take place at night [2]. Pedestrian fatalities that occur at night result from such factors as poor visibility, drivers fatigue, driving speed, and alcohol [3].

In view of the above problems, many organizations set up preventive measures. With the efforts undertaken by the European Union (e.g., the “Road Safety Program” [4]), the total number of fatalities in car accidents is falling rapidly. It changed from 54,000 in 2001 to 31,000 in 2010 [4]. The number of pedestrian-related accidents was 9,100 in 2001 and 5,500 in 2010. This represents a global downward trend in the average pedestrian mortality across the European Union, but some exceptions are also noted [2]. In some countries, especially those of rapid economic growth, e.g., in Poland and Romania, this trend is somehow weaker, i.e., in Poland there were 1,866 pedestrian fatalities in 2001 vs. 1,236 in 2010 [2].

Thanks to new achievements in the technological sciences, it is now possible to offer tools that can aid transportation safety. In the automotive-related areas, it could be found such mechanisms as road planning, road security, assisting of drivers and their capabilities, protection of drivers and passengers, protection of pedestrians, and many others. Automotive companies offer advanced driver assistance systems (ADAS) solutions that increase the safety of night traffic. Among the most popular are: adaptive (intelligent) front lights, detection of weariness or intoxication of a driver, warning of lane departure, recognition of traffic signs, information of a vehicle blind spot, automatic braking (the last one typically works under the limited speed and is dedicated to the city limits and traffic jams).

The car manufacturers also offer ADAS for night-vision. Such systems can improve driver perception by offering more time to react. By this, they protect against accidents with pedestrians, who are, in fact, defenceless in contact with vehicles. The first night-vision system has been introduced to the market by General Motors in the year 2000 and applied in the Cadillac DeVille. The development of this project took 15 years of 70 person team and cost approximately $100 million [5]. In 2003, Toyota has introduced the first commercial active night-vision system for Toyota Landcruiser and Lexus LX470 and reached the range of 100 m. In 2004 Honda has introduced it in the Legend model as an optional system named "Intelligent Night-vision" with the first option of pedestrian detection. The system gained a range between 30 and 80 m [6]. Nowadays, such systems are offered by most car manufacturers on the market, but they are still dedicated to the premium level cars.

The night-vision systems can be classified twofold: as passive or active systems, taking image acquisition methods into account. The passive systems capture far-infrared, thermal radiation (thermo-vision) naturally emitted by any object with a temperature above absolute zero. In contrast, the active systems are equipped with near-infrared illuminators and capture the light reflected from the objects.

In passive systems, each object with a temperature greater than 0 K emits radiation, but in practice, only the objects other than the surroundings become distinctive. Good contrast for living beings is one of the significant advantages of the passive systems. A range of detection is much more extensive than in active systems. For high-quality cameras, it can reach 300 m. Thermal imaging cameras are also not blinded by the lights of other vehicles. This feature is significant because it does not cause the distraction of a driver.

With night-vision, it is possible to set up pedestrians detection feature for night-time driving. This feature is essential in autonomous vehicles that are starting to appear on roads worldwide. However, the detection of pedestrians is a challenging task. In general, there are many approaches to solve this goal. A natural choice is a vision because it is based on how people perceive humans. The solutions can be divided into classic monocular vision systems [7], stereo vision systems [8], and infrared vision systems [8–10]. Moreover, the vision systems are relatively inexpensive and easy to interact with humans (drivers).

More advanced arrangements use ultrasonic sensors [12], conventional radar [13], or LIDAR (Light Detection and Ranging) to retrieve a 3D map of the terrain and detect pedestrians [14]. Nowadays, these sensors are often used simultaneously with vision sensors (in a sensor fusion manner) in autonomous vehicles to increase the accuracy of detection of objects, pedestrians, and threats on the road [15], [16].

The vision-based pedestrian detection systems usually perform in four main stages (see Figure 1): first, the image acquisition, second, preparation of the so-called region of interest (ROI), which separates objects of interest from the background for further processing, third, the object classification, which distinguishes pedestrians from other objects.

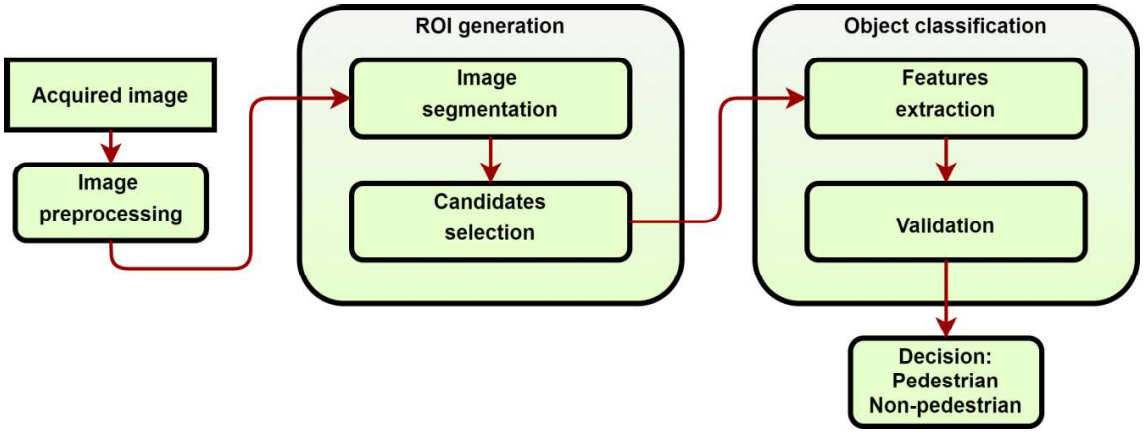


Figure 1. The general pedestrian detection procedure

In the case of pedestrian detection with moving camera, i.e., in cars or autonomous vehicles, the detection process should be both accurate and computationally efficient to enable detection of pedestrians in a real-time manner.

For this reason, the pedestrian detection process should be well optimized at the ROI generation and object classification stages. In the case of ROI generation, the choice of segmentation technique significantly impacts accuracy and computational efficiency.

Currently, in the case of pedestrian detection on thermal images in night conditions, similar image segmentation techniques are used as for color day-time imaging cameras, i.e., sliding window-based techniques [11], [17], or region proposal neural networks [18]–[20]. These techniques do not directly use information about thermal contrast (pedestrians are usually brighter than their surroundings at night). Moreover, most of these solutions require highly efficient hardware, i.e., GPGPU for real-time operation, making it difficult to use in vehicles.

It is potentially possible to perform segmentation by thresholding. This approach allows for a significant acceleration of the entire pedestrian detection process by reducing the ROIs area in the image. It uses properties of the thermal images (the pedestrians are usually warmer, therefore brighter than the surroundings). So far, several techniques for the segmentation of thermal images based on thresholding have been proposed [7], [21], [22]. However, these techniques are currently not widely used due to lack of the state-of-the-art accuracy of pedestrian segmentation and operational stability.

Figure 2. Illustrative example of infrared image with two pedestrians

The simple assumption that pedestrians are warmer than the surrounding at night is not always valid. Many problems arise during segmentation, i.e., the uneven level of the observed temperature of one pedestrian (see Figure 2) and the temporary loss of thermal contrast between the pedestrian and the surroundings. All the above-mentioned problems should be compensated to avoid the situation that a pedestrian is not being detected at the segmentation stage. This raises the question of whether accurate and stable pedestrian segmentation of thermal images through thresholding is possible.

In the case of the object classification stage (see Figure 1), the selection and tuning of the appropriate technique also have a significant impact on overall detection

efficiency. Pedestrian detectors often use ready-made solutions as the so-called black boxes by scaling the resolution of the recorded image to the resolution of the used object detector [11], [23]. This is especially often practiced with deep convolutional neural networks [24], [25]. Therefore, the computational performance of these detectors is often very low and requires powerful hardware for real-time operation. For this reason, it is essential to properly fit the algorithm to the image source properties – i.e., to the sensor type, camera perspective, and resolution of the image in order to increase the computational efficiency of the pedestrian detector without affecting the accuracy.

1.2. Aim of work and scientific thesis

The initially described problems in previous Subsection 1.1 concern the process of pedestrian detection: the possibility of using thresholding for accurate and efficient ROI generation of thermal images at night and the necessity of proper tuning of the object detection stage. These are important scientific problems, thus they require further analysis and research.

The scientific aim of this Ph.D. dissertation is the analysis and development of automated mechanism for highly efficient night-vision pedestrian detection on thermal images. The research presented in this dissertation is focused to two main issues: ROI generation based on thresholding and procedure of tuning object classification stage. The author's motivation was to achieve the state-of-the-art accuracy and real-time performance of pedestrian detection process in order to apply it in vehicles (such as ADAS equipped cars or autonomous vehicles) without using special hardware i.e., general-purpose computing on graphics processing units (GPGPU).

The scientific thesis can be formulated as follows: The developed approach of night-vision pedestrian detection based on proposed ROI generation by thresholding of thermal images and by properly tuned object classification procedure improves detection accuracy and significantly increases computational efficiency of the pedestrian detection process.

1.3. Main scientific achievements

The main scientific achievements presented in this dissertation are innovative modifications of the night-vision pedestrian detection process. They can be divided into three main groups:

• a new ROI generation approach for the thermal images based on image thresholding,

• a technique of additional ROI adjustment (slightly enlarging the ROI area of the image) before the object classification stage,

• a proposition of procedure for tuning of object classification process with the universal performance index.

The complete structure of the proposed pedestrian detection algorithm with the introduced improvements is presented in the diagram in Figure 3 (cf. Figure 1 for the author's improvements).

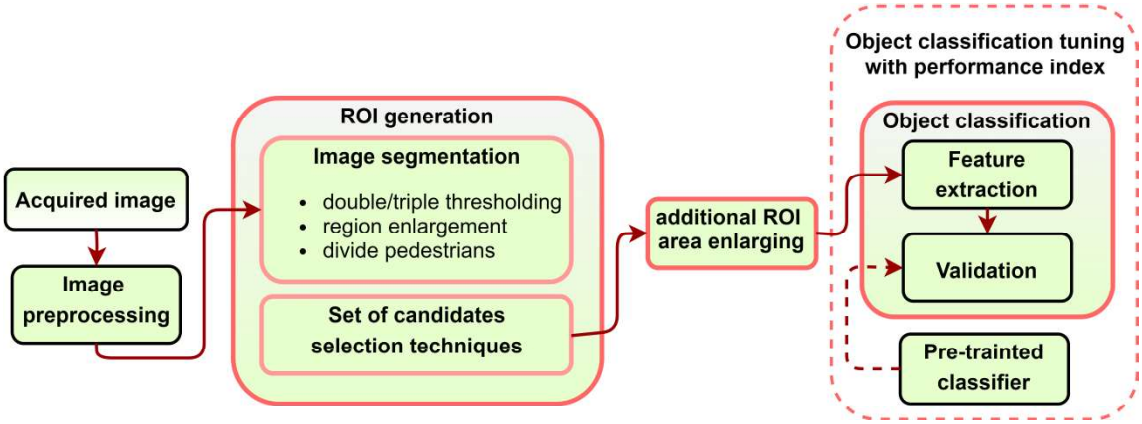


Figure 3. The pedestrian detection scheme with proposed improvements (cf. with Figure 1)

At the ROI generation stage, the author proposed the image segmentation technique of thermal images by multiple thresholding with two or three global thresholds. Then, to compensate the imperfections of the thresholding process and to increase the accuracy, the techniques of regions enlargement and dividing wide ROIs were proposed. Moreover, in order to effectively accelerate the proposed ROI generation procedure, a set of candidate selection techniques was proposed.

The technique of additional ROI area adjustment was proposed to increase the accuracy of the object classification stage. It consists of analysing proportionally larger areas from the image than those detected after the ROI generation stage. This technique allows for increasing the accuracy of the entire pedestrian detection algorithm with a negligible impact on processing time.

The specialized procedure of tuning the object classification stage also was proposed to adjust the detector parameters. This procedure is based on a novel and universal performance index. Using this procedure, the author demonstrates that properly tuning of the object detection stage to the analysed image source properties - e.g., to the sensor type, camera perspective and resolution of the image is important and significantly affects the computational performance. The author proved that it is possible to significantly reduce the processing time without affecting the accuracy. Moreover, the

presented approach is quite general, i.e., it may be applied not only to the considered problem but it can be adapted to detection of any type of object with any classifier.

Finally, the proposed improvements made it possible to propose an efficient pedestrian detection algorithm for thermal images in night conditions. The very high computational efficiency of detection process was obtained, with up to 130 frames per second using the CPU only. Moreover, it was possible to obtain the state-of-the-art detection accuracy for tested detectors, namely the aggregated channel feature (ACF) and deep convolutional neural network (CNN). This was confirmed by the sets of experiments performed on two public benchmarks, i.e., CVC-14 [23] and KAIST [26].

The structure of this dissertation is as follows: after the introduction, in Chapter 2, the extended analysis of the research area (initially described in Chapter 1.1) is presented along with a summary and detailed explanation of the motivation to undertake this work.

Section 2.2 presents description of all the public datasets of night-vision recordings used in the experiments. Subsequently, the proposed improvements to the pedestrian detection process are presented in separated chapters.

The proposed ROI generation approach is described and tested in Chapter 3. In the next Chapter 4, the problem of inaccurate matching of the edges of ROI to the outer edges of the pedestrians in the image was analysed with the proposed additional ROI area adjustment technique. Chapter 5 presents and describes the proposed procedure of tuning the object classification stage with universal performance index.

Chapter 6 presents the experiments on the proposed pedestrian detection algorithm performed to compare the presented solution with the standard approaches based on the sliding window segmentation technique and other solutions in the literature. This chapter also presents software with multi-threaded architecture. Chapters 3 and 5 also include separate experiments and tests performed to verify the effectiveness of the proposed modifications.

In the last Chapter 7, the author also presents additional research on the possibility of using so-called multi-spectral vision for scene analysis by monitoring operators. It concerns thermo-vision merged with a standard camera as an option for CCTV monitoring. Performed experiments show that this option shortens reactions and supports faster identification of objects (e.g., pedestrians) at night. The dissertation is closed with the conclusions.

2. State of the art

This chapter contains an extended analysis of the research area of night-vision pedestrian detection. Firstly, the two basic night-vision image acquisition techniques are discussed: near-infrared vision and far-infrared vision (thermal imaging) with their advantages and disadvantages. Secondly, the analysis of existing approaches to ROI generation is presented. Thirdly, the methods of object classification for final ROI verification are discussed. In the last subsection, the necessity of introducing more efficient and more adapted to image source methods is indicated.

2.1. Night-vision approaches

The night-vision systems can be classified twofold: as passive or active systems, taking image acquisition methods into account. In both approaches, the infrared band of an electromagnetic radiation is used.

There are several conventional divisions of the infrared band into sections depending on the application and sensors sensitivity range [27]–[31]. One of the detailed infrared division is presented in [29]:

- Near-Infrared (NIR) with wavelength from 0,7 μm to 1,4 μm,
- Short-Wave Infrared with wavelength from 1,4 μm to 3 μm,
- Mid-Wave Infrared with wavelength from 3 μm to 8 μm,
- Long-Wave-Infrared (LWIR) with wavelength from 8 μm to 14 μm,
- Very Long-Wave-Infrared with wavelength from 14 μm to 25 μm,
- Far-Wave-Infrared with wavelength from 25 μm to 1000 μm.

In the field of Intelligent Transportation Systems (ITS) and automotive night-vision, the term Far Infrared (FIR) is used concerning the LWIR band and thermal imaging [11], [23], [27], [30]–[32]. Therefore, in this work, the term FIR is also used to define the LWIR infrared band and refer to the range of sensitivity of thermal imaging cameras.

In general, the passive systems capture FIR, thermal radiation (thermo-vision) naturally emitted by objects, while the active systems are equipped with NIR illuminators and capture the invisible to the human eye NIR light reflected from the objects (see Figure 4).

In the case of active NIR systems, very often the typical silicon-based digital sensors are used. They are sensitive not only to the visible-light spectrum in the range of 400-700 nm but also to the NIR range [28], [31], [33]. These cameras are typically used with a permanent IR cut-off filter in good lighting conditions. However, the CCTV day and night (visible/NIR) cameras use a mechanical IR filter that switches depending on the time of day. Since NIR imaging does not require significant investments (only additional NIR illuminators), it is commonly used in stationary CCTV systems for night-vision, e.g., in security applications.

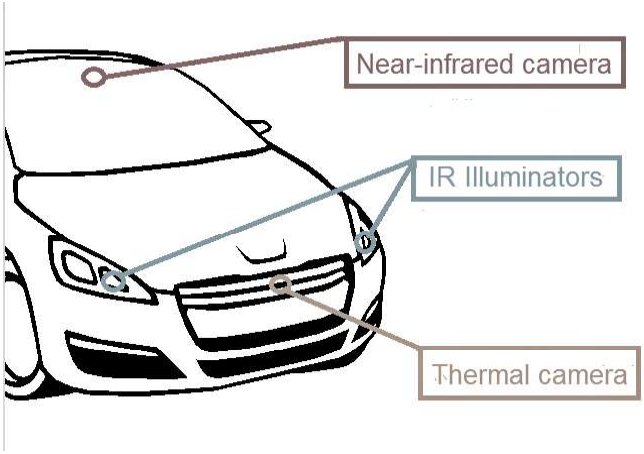


Figure 4. Typical location of components for NIR and FIR systems on the example of the automotive night-vision system

In passive systems, the thermal imaging camera captures FIR radiation emitted by objects. In the FIR range, the radiation power of objects in the environment depends on their temperature. Each object with a temperature greater than absolute zero emits radiation, but in practice, only the objects with temperatures other than the surroundings become distinctive [31]. As a result, the internally heated objects such as pedestrians, cars in motion (with engines, radiators, heated reflectors) are clearly visible (see Figure 5).

In general, two types of thermal detectors are used in thermal cameras: photon detectors and thermal detectors [27], [31]. Photon detectors are based on photoeffect. The absorption of photons in the material causes the emission of electrons that change the current flowing through the detector. In the case of thermal detectors, the absorption of FIR radiation changes the temperature of the detector, which causes a change in electrical properties: electrical resistance in the case of microbolometers or electric polarization in the case of ferroelectric detectors.

Nowadays, microbolometer detectors have been increasingly used. These detectors do not require refrigeration, which makes them more compact and reduce the price. The microbolometers cameras that are used for people detection typically use a range of 7 - 14 μm [27], [29], [31].

The precise remote temperature measurement is very difficult to achieve. The accuracy could be influenced by many factors, including the emissivity of the material from which the object is made, the surrounding atmospheric conditions, i.e., fog or rain, the distance from the camera, the transmission of radiation through the atmosphere [34]. Therefore, to precisely determine the temperature value on a given surface, it is necessary to know many environment variables. This is well illustrated by the formula for the total radiation power received by the detector [35], [36]:

$$W_{\mathrm{tot}} = \varepsilon \cdot \tau \cdot W_{\mathrm{obj}} + (1-\varepsilon) \cdot \tau \cdot W_{\mathrm{amb}} + (1-\tau) \cdot W_{\mathrm{atm}}, \quad (1)$$

where: ε is the emissivity of the object, τ is the transmission through the atmosphere, $(1-\varepsilon)$ is the reflectance of the object, $(1-\tau)$ is the emissivity of the atmosphere, W_{obj} is radiation power of the observed object with a temperature T_{obj}, W_{amb} is

radiation power reflected from the object and generated by ambient sources with temperature T_{amb}, W_{atm} is radiation power from the atmosphere with temperature T_{atm}.

In thermal imaging for the application of pedestrian detection, the thermal contrast between objects and the environment is the most important feature since the detector uses shapes in the image to classify the objects. Additionally, the difference between the emissivity of the materials may result in additional, artificial edges in the image.

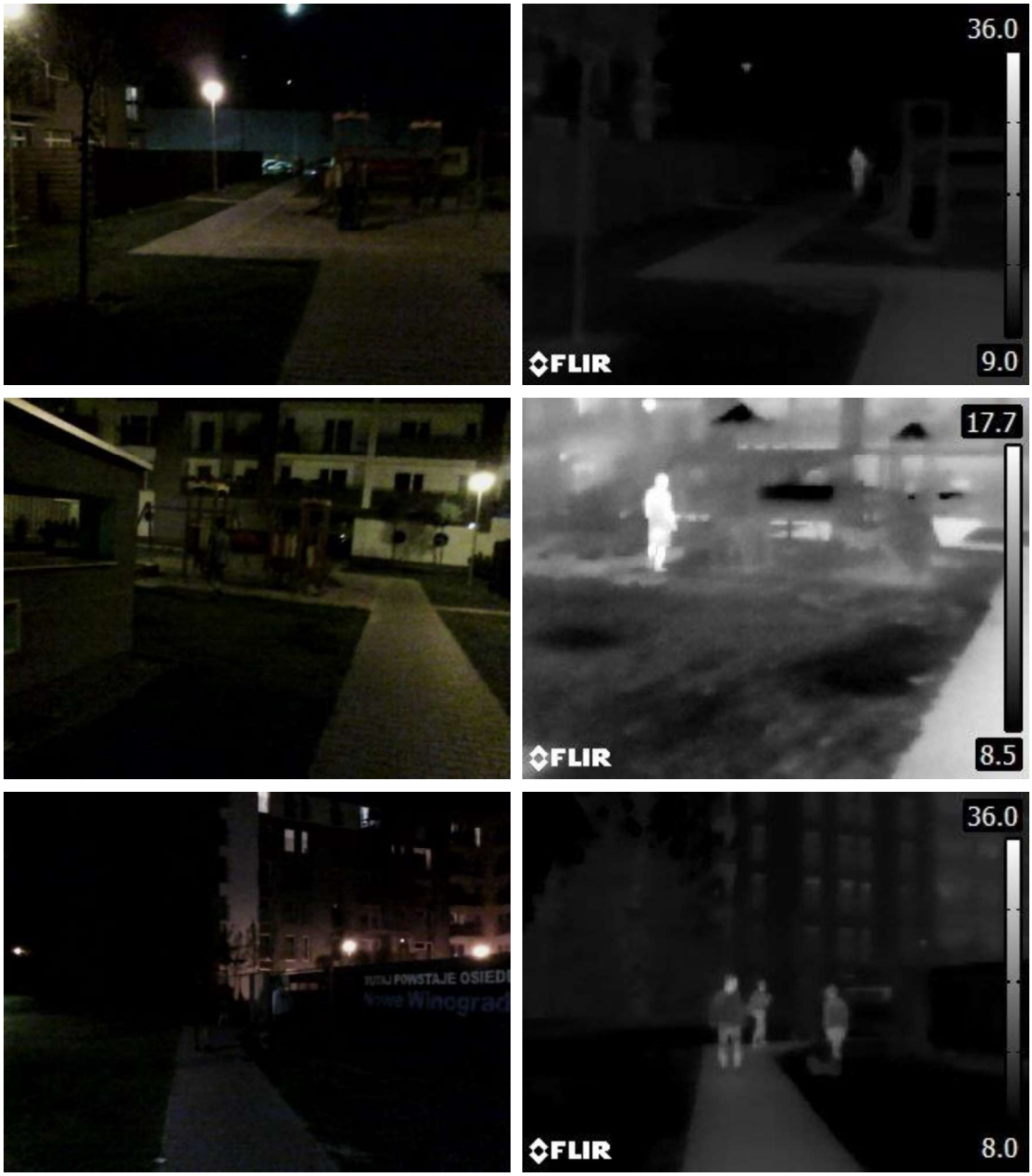


Figure 5. Illustrative examples of visible color images (not NIR, first column) with corresponding thermal images recorded at night (marked temperature scale, second column)

2.1.1. Comparison of NIR and FIR systems

The high contrast between living beings and their surroundings is one of the major advantages of passive systems. A range of detection is much more extensive than in active systems and for high-quality cameras, it can even reach 300 m at night for a standard camera. Another important factor is that thermal imaging cameras are not blinded by the lights of oncoming vehicles [30], [31].

The main disadvantage of the passive thermo-vision comes from the physical basis of this type of imaging: the measured emission of an object strongly depends on the source material and the covering of the object. It makes the calibration of the system difficult and strongly context-dependent. Fortunately, the absolute calibration of the camera in automotive night-vision applications is not as important as, e.g., for the typical thermal imaging in the construction industry.

Among other disadvantages of thermal cameras are lower resolution and higher costs than for the cameras used in the active systems. Because of a specific way of image capture, they are characterized by a weak representation of the textures and low signal dynamics (as presented in Figure 6) [30], [31]. Additionally, the FIR spectrum is more difficult to interpret for a driver: e.g., tires are white (hot), and the rest of the car is black. Other, typically high-contrast objects like horizontal lane markings or headlamps (LED or rear lights) are not visible in the image. Another significant disadvantage is the sensitivity to changes of thermal contrast: with the season, weather, humidity.

The main advantage of the active systems is high resolution. The image is easy to interpret for the driver because of the proximity of NIR to the visible light (see Figure 6). It is possible to, e.g., see the lanes and the headlights of oncoming vehicles. The relatively low cost of NIR cameras and their small size makes them attractive and widely available. The cameras of this type can also be used in other systems and successfully work in day light (e.g., CCTV cameras are often equipped with the mechanically switched IR filter used as a day/night switch).

The active systems have a shorter detection range than their passive counterparts and reach about 150 m. This distance strongly depends on the power of illuminators. However, typically this disadvantage is compensated by a higher resolution of image sensors. The NIR detectors can also be dazzled by the headlights (or illuminators) of oncoming vehicles and operate significantly worse than the FIR cameras in fog. The advantages and disadvantages for both systems are summarized in Table 1 (where "+" denotes a slight advantage, and "++" denotes a significant advantage between NIR and FIR systems).

Finally, both active and passive systems are used in various applications. Active systems are cheaper and have better resolution than passive ones, but pedestrian detection needs more complicated algorithms.

The videos obtained by these two types of systems differ a lot, and thus the video processing algorithms should be optimized separately for each of these two types.

Figure 6. Illustrative examples of NIR images (first column) recorded at night from TetraVision [37] dataset with corresponding FIR images (second column)

Table 1. Summarized advantages and disadvantages of NIR and FIR night-vision systems

FIR	Feature	NIR
	Image quality *(resolution, textures)*	++
+	**View range** *(range, angle)*	
++	**Thermal contrast of living beings** *(contrast to the background)*	
++	**Dazzling effect** *(temporary blinding by oncoming vehicle)*	
+	**Ability to operate in difficult conditions** *(fog, rain)*	
	Assembly and maintenance *(system integration, calibration)*	+
	System price *(camera, components)*	+

2.2. Night-vision datasets

This section describes all the night-vision datasets that were used in the experiments presented in this dissertation. These datasets are known in the analysed area of research and were commonly used for benchmark tests in many papers, e.g., [26], [32], [38], [39]. They are: CVC-09 (Computer Vision Center, FIR Sequence Pedestrian Dataset) [40], CVC-14 (Computer Vision Center, Visible/FIR Day/Night Sequence Pedestrian Dataset) [23], NTPD (Night-time Pedestrian Dataset) [41], LSI FIR (Laboratorio de Sistemas Inteligentes, Intelligent System Lab Far Infrared Pedestrian Dataset) [27], OSU (Ohio State University, Thermal Pedestrian Dataset) [42], KAIST (Korea Advanced Institute of Science and Technology, Multispectral Pedestrian Detection Benchmark) [26].

Table 2. Number of training and testing samples in night-vision datasets used for experiments with object classification stage (in Chapter 5)

	No. of training samples		**No. of testing samples**	
dataset	**positive samples**	**negative samples**	**positive samples**	**negative samples**
CVC-09 FIR Day-time	11,839	25,410	6711	75,398
CVC-09 FIR Night-time	6998	30,030	7862	72,985
Extended NTPD	1998	8730	2370	12,600 (*)
LSI FIR	10,208	43,390	5944	22,050
OSU	1004	1932	964	1932

Tested datasets differ in resolutions, quality, and acquisition techniques. The CVC-14 and KAIST datasets were used in the experiments with the proposed ROI generation approach (in Chapter 3) and the final experiments with the proposed pedestrian

detection procedure (Chapter 6) because they had images of the FIR spectrum and it was possible to compare the achieved results with the literature. The other datasets (OSU, NTPD, LSIFIR, and CVC-09) were used with the object classification stage (in Chapter 5). For all of them, ROI samples were extracted for both training and testing (Table 2).

2.2.1. CVC-09 Thermal Pedestrian Dataset

The CVC-09 (Computer Vision Center, FIR Sequence Pedestrian Dataset) consists of two subsets of pedestrian thermal images: 5990 images recorded during the day and 5081 recorded at night. Their resolution is relatively high as for the FIR recordings and equals 640×480 pixels. The authors of this dataset inform that it was produced with the FIR thermal imaging technology. However, they do not specify the camera type and the temperature scale [40]. The images have some unknown static temperature scale, and there is no contrast enhancement applied.

This dataset is very demanding as pedestrians occur with various sizes. Images recorded on days have low contrast between pedestrians and the background. This differs from other typical FIR recordings.

The dataset with positive samples was prepared by clipping pedestrians out of the original images (see Figure 7). The resulting dataset was annotated automatically. Therefore, there are some inaccuracies, e.g., not all pedestrians were correctly marked (cf., Figure 7b - a figure in the third column contains parts of two pedestrians).

Figure 7. CVC-09 dataset of pedestrians: (a) day-time positive samples, (b) night-time positive samples, (c) day-time negative samples, (d) night-time negative samples.

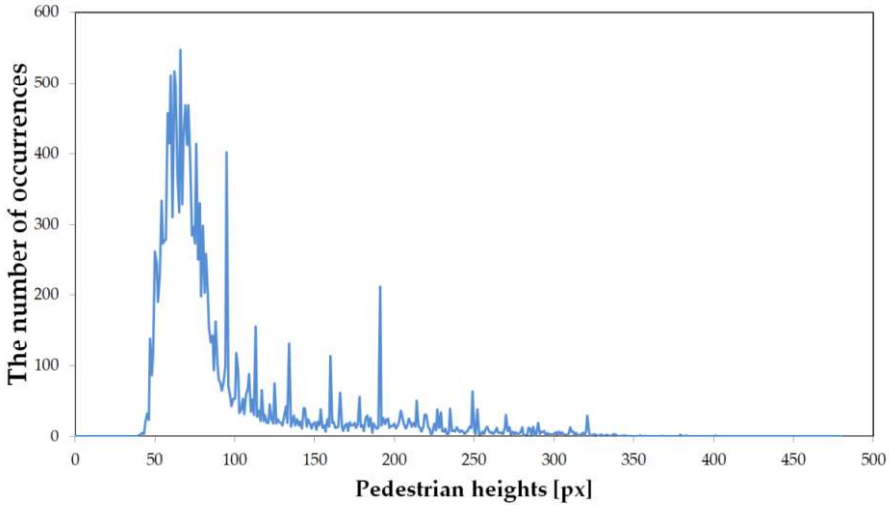


Figure 8. Distribution of pedestrian heights (in pixels) in CVC-09 dataset

Due to the variety of distances between the camera and pedestrians, the obtained positive samples have different resolutions (from 3×6 up to 190×458 pixels). The height distribution of these samples is shown in Figure 8. Because all samples have to be scaled to a given classifier resolution, they sometimes must be significantly enlarged (up-scaled), and then they can be quite strongly blurred (cf., Figure 7a, 7b).

The dataset with negative samples was prepared by cutting out chosen areas with no pedestrians. They were extracted with a window size equal to the largest used classifier resolution (i.e., to 64×128 pixels). During the classifier training, the negative samples were then scaled down again to the required resolution. The prepared dataset is large enough for statistical analysis.

2.2.2. CVC-14 Visible/FIR Day/Night Sequence Pedestrian Dataset

The CVC-14 dataset contains of multimodal (FIR plus visible) video sequences [23]. This dataset is divided into two subsets: recordings captured by the FIR camera in a day and at night. However, only night FIR recordings were used (see Figure 9). The CVC-14 dataset is very demanding for testing of automatic image processing procedures: pedestrians have various sizes, images are of low contrast between pedestrians and the background. This is mainly due to the time and place of recordings – hot summer in Spain. As a result, there are many hot regions, which had been heated up during the day. Despite the drawbacks, the dataset was selected because it presents pedestrians in different scales and enables extracting pedestrians straight from the images using enough accurate ground truth.

For the training of classifiers, positive datasets of samples were prepared (images) by cutting out pedestrians from the original frames (Figure 10). The negative samples were prepared by cutting out areas, which do not contain pedestrians. They were extracted by a window of size of the largest used classifier (i.e., 64×128 pixels).

The details about the training and testing sets of samples are presented in Table 3.

Table 3. Training and testing subsets extracted from night-time CVC-14 FIR pedestrian dataset

CVC-14 night-time FIR dataset		**No. of samples**
Training subset	Pedestrian samples	2222
	Negative samples	10,242
Testing subset	Positive frames	703

Figure 9. Four illustrative images from the CVC-14 dataset

Figure 10. Illustrative examples of pedestrian and non-pedestrian samples from the CVC-14 night-time dataset

2.2.3. Night-time Pedestrian Dataset

The NTPD (Night-time Pedestrian Dataset) [41] is divided into two sub-sets: training and testing (details are presented in Table 2). It consists of images of pedestrians stored with the NIR active system of resolution 64×128 pixels (cf., Figure 11). In this dataset, to make the classification process realistic, the number of the negative samples was extended similarly to those occurring in real situations of the automotive applications as an asymmetric distribution (much more negative samples than the positive ones) is quite typical. These negative samples were extracted from images, which contain no pedestrians.

Figure 11. Pedestrian (positive) samples from NTPD dataset

2.2.4. LSI FIR Pedestrian Dataset

In the LSI FIR (Laboratorio de Sistemas Inteligentes/Intelligent System Lab Far Infrared Pedestrian Dataset) [27], the FIR images were acquired in outdoor urban scenarios. The images are divided into two subsets: the classification dataset and the detection dataset. The first one is divided in a train and a test sets. The train set contains 10208 positives and 43390 negatives, while the test set contains 5944 positives and 22050 negatives (as presented in Table 2). The images are scaled to 32×64 pixels and include positive and randomly sampled negative images. The detection dataset includes annotated original positive and negative images of 164×129 pixels resolution. In the experiments, only the first subset was used.

2.2.5. OSU Thermal Pedestrian Dataset

The OSU (Ohio State University) Thermal Pedestrian Dataset consists of 10 daytime video sequences captured on a university campus under various weather conditions (cf., Figure 12). These sequences were recorded using a passive thermal sensor Raytheon 300D [42]. Thus, the images have a resolution of 320×240 pixels.

Based on this dataset, several authors created their own, not standardized training and test subsets [32], but with a small number of samples. Since pedestrians in the original dataset have low resolution, it was decided to extract samples with a resolution of only 32×64 pixels. From half of the images, pedestrians were selected who, together with their mirror images (used to increase the number of samples), formed positive training samples. From the second half of the images, in the same way, the training samples were created. To obtain negative samples (those without pedestrians), frames with the background only were cut with a window of size 32×64 pixels with a spacing of 8 pixels.

Additionally, their number was increased by rotation and mirroring vertically and horizontally. Finally, 3864 negative samples were obtained. Half of them were used for

training and the other half for testing. The extended version of this dataset is also available on [42].

Figure 12. Two illustrative images from the OSU dataset

2.2.6. KAIST Multispectral Pedestrian Detection Benchmark

The color-thermal KAIST dataset contains 95,328 aligned color-thermal image pairs, with 103,128 dense annotations on 1,182 unique pedestrians. This dataset was recorded with the PointGrey Flea3 color camera with a resolution of 640×480 pixels and a 103.6° vertical field of view, and the FLIR-A35 thermal camera with a resolution of 320×256 pixels and a 39° vertical field of view (see Figure 13). The thermal image was aligned with the color image by cropping an area of the color image. This dataset provides 20 frames per second. Details of the selected sequences are presented in Table 4.

The dataset consists of 12 (6 train and 6 test) image sequences recorded day and night and in different areas (campus, road, and downtown, as presented in Table 4). In the experiments, only thermal images captured at night were used.

This dataset is provided with the manually annotated, detailed ground truth for each image frame. Annotations contain information about the pedestrian's position in the image, the distance from the camera (or the size in the image), and scale: near, medium, or far. Pedestrians are also marked with one of three occlusion tags: no occlusion (78.6%), partial occlusion (12.6%), and heavy occlusion (8.8%).

Table 4. Details of the selected sequences from KAIST dataset (only night-time recordings were used in experiments)

Sequence	Area	Frames	Pedestrians
Set 03 (train)	campus	6 668	7 418
Set 04 (train)	road	7 200	17 579
Set 05 (train)	downtown	2 920	4 655
Set 09 (test)	campus	3 500	3 577
Set 10 (test)	road	8 902	4 987
Set 11 (test)	downtown	3 560	6 655

Figure 13. Four illustrative images from the KAIST dataset (original images with increased brightness and contrast)

2.3. General procedure for pedestrian detection

The typical scheme of the procedure for pedestrian detection is presented in Figure 1 (the most general scheme), in Figure 3. The pedestrian detection scheme with proposed improvements (cf. with Figure 1), and in Figure 14 (a more detailed scheme with the indication of the most important approaches). Its first stage is the IR image acquisition and then image preprocessing. For image preprocessing standard techniques are used in order to reduce noise and enhance image contrast.

In the second stage, the ROIs are generated, covering all areas with pedestrian candidates for further processing. The first step in ROI generation is image segmentation performed to separate pedestrians from the background (or more precisely, the desired areas of the IR image that potentially contain pedestrians called pedestrian candidates or ROIs). Correctly segmented ROIs contain all objects to be detected (pedestrians), but together have as few other objects (none pedestrians) as possible. By such means, the amount of data that is transferred to the next stages is reduced. There are plenty of solutions for proposing the pedestrian candidates, starting from the sliding window approach in a multi-scale manner [11], [23] up to faster and intelligent solutions [10], [43], [44], e.g., the specialized region proposal networks [45], [46].

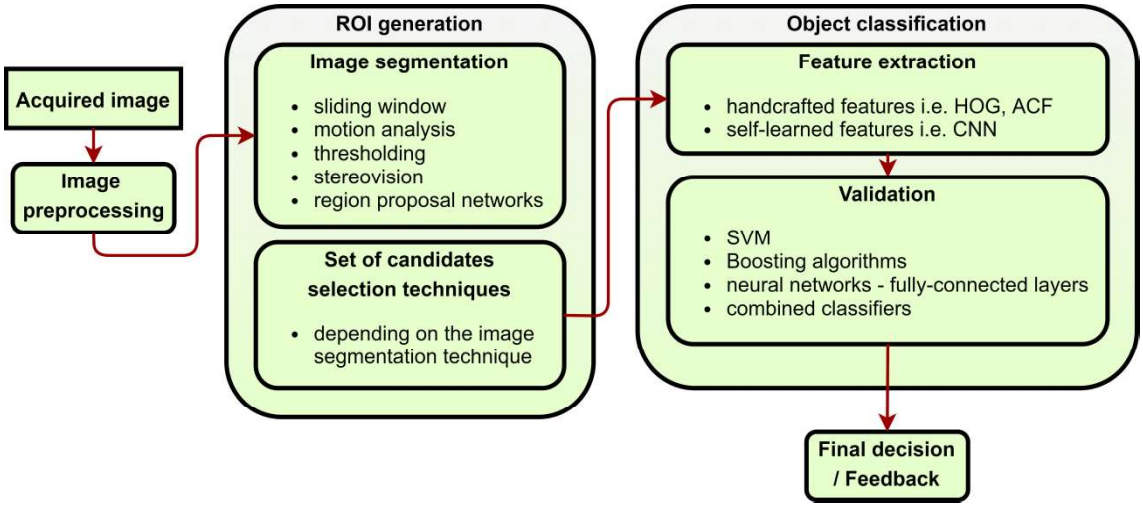


Figure 14. The general scheme of the pedestrian detection procedure with listed most important approaches at each step

After the ROI generation, the next is the pedestrian classification stage. This is a crucial stage as it strongly affects the final quality of pedestrian recognition. The object classification stage consists of two steps: feature extraction and final validation with the selected classifier. The feature extraction step brings the most valuable features and reduces the amount of data that describes the object. In a validation step, the classifier finally decides which objects are pedestrians and which are not.

2.4. Region of interest generation

Image segmentation, which is also referred to as foreground segmentation or candidate generation, extracts the so-called regions of interest from the image avoiding as many background regions as possible. The segmentation is of remarkable importance not only to reduce the number of candidates but also to avoid scanning outer regions like the sky. This stage cannot miss pedestrians, otherwise, the consecutive modules, e.g., the classifier will not be able to correct the failure.

The image segmentation used for pedestrian detection very often includes a selection of candidates by pedestrian size constraints. These constraints refer to the aspect ratio, size, and the position that candidate ROI must fulfill to be considered as a pedestrian [7], [32].

This section provides an analysis of possible approaches to ROI generation. Then, two basic thresholding techniques are presented: the global Otsu threshold and the locally adaptive dual-threshold procedure. These algorithms are also discussed in the next chapter, therefore a more detailed description is included in the following section.

2.4.1. Analysis of image segmentation approaches

Sliding window

A sliding window technique, as exhaustive scanning approach [11], [23] belongs to the simplest segmentation procedures. The sliding window selects all possible candidates in the image according to the pedestrian size constraints without explicit segmentation. To find pedestrians at different sizes, the scanning window must be scaled down (or up) after each scan.

The sliding window procedure has two drawbacks: it is very time-consuming and produces a large number of candidates, which increases the potential number of false-positive decisions.

An interesting solution to these problems is presented in [47]. The method uses a key point-centric sliding window with a classifier. In [17], the sliding window operates on the preprocessed image with luminance saliency, sharpening the edges. It also uses energy symmetry to speed up calculations.

Another acceleration method [48] uses Markov Chain Monte Carlo sampling to estimate the probabilistic density distribution of the classifier responses. Then the search strategy can be adjusted according to the distribution. In [49], the step of the sliding window is filtered on the optical flow images.

Stereovision

A stereovision was also proposed for IR pedestrian detection, e.g. in [8], [9], [11], [50]. Stereo systems offer robust detection with such techniques as disparity map or histogram and can be used to effectively find ROI [11]. However, at least two thermal imaging cameras are not a viable option for many automotive designers as costs, power consumption, and physical space are significant factors.

In the Protector system [51] the returned map is multiplexed into different discrete depth ranges, which are then scanned with the window according to the pedestrian size

constraints, taking into account the location of the ground plane. If the depth features in one of the windows exceed a given ratio, the window is passed. Otherwise, it is canceled. Some authors [9], [50] use the v-disparity representation [52] to identify the ground and vertical objects. Like consumer monocular FIR cameras, other papers combine a visual sensor and FIR, but in a stereovision manner. This approach, which corresponds to sensor fusion, is worth mentioning because of its potential to widen the range of working conditions, i.e., both in the daytime and night-time.

In [8], single thresholding by entropy maximization is performed, but the next step is a disparity map calculation (from stereovision), which finally improves the results.

Motion detection

A motion feature in video processing contains practical and discriminative information. Inter-frame motion and optical flow may be used for foreground segmentation, primarily in the general context of moving obstacles detection [53].

In [54], the histograms of oriented gradients (HOG) feature on the optical flow images was computed to get the Histogram of Oriented Flow feature. In [55], the pedestrian motion information was utilized within an generalized expectation-maximization framework to generate the candidate pedestrians.

However, the motion-based segmentation requires a fixed position of the camera, limited background motion, and does not detect standing pedestrians.

Region proposal networks

Another group of image segmentation techniques is the region proposal networks (RPN) [56], which are based on CNN. The vast majority of these currently developed techniques are dedicated to color images [20], [45], [46], [57]. However, there are also some implementations for FIR [18], [19] and multi-spectral imaging [56].

R-CNNs were initially developed in [57]. The high-capacity convolutional neural networks were applied to bottom-up region proposals in order to localize and segment objects. The Fast R-CNN proposed in [20] improves training and testing time with a single-stage training and accelerated fully connected layers. The Faster R-CNN [45] improves detection time even more by sharing full-image convolutional features with the detection network to improve the time efficiency of pedestrian detection. Another solution, YOLOv3 (You only look once) [46] apply a single neural network to the full image. This network divides the image into regions and predicts bounding boxes and probabilities for each region.

The authors of [19] proposed to augment thermal images with their saliency maps and applied them to Faster R-CNN. In [18], a pre-trained YOLOv3 pedestrian detector is adapted to detection in the thermal-only domain generative with a data augmentation strategy.

In general, RPN are very accurate but they are also computationally demanding and need powerful hardware, e.g., graphic processing units (GPUs) or tensor processing units (TPUs).

Thresholding techniques

The last group of ROI generation techniques are techniques based on image thresholding. They are designed mainly for FIR images, because pedestrians at night conditions are usually warmer and hence appear brighter than the background in FIR images. The simple thresholding of the image is a common starting point for extracting pedestrian candidates. A brightness threshold is selected that separates the foreground from the background. Then, each pixel is classified according to the selected threshold value.

The global threshold (calculated once and used for all pixels) can be calculated as an average value of the difference between the maximum and the minimum image intensities. The method presented in [21] defines a bright pixel threshold as the difference between the maximum image intensity and a given constant. A threshold value is defined from the mean and the maximum image intensity values [22]. In [58], a threshold value is chosen as the last local minimum of the image histogram before the saturation point.

In [59], a static threshold is derived with the Bayes classifier performing on a set of templates known to contain pedestrians.

A well-known adaptive method, called Otsu's method after the inventor's name [60], belongs to the clustering-based image global threshold methods. It assumes a bi-modal histogram (see Figure 15) with foreground pixels and background pixels and finds the optimum threshold separating these two classes.

More advanced threshold methods are based on two thresholds. A region-growing style threshold using two static thresholds is implemented in [61]. The lower threshold is restricted to areas spatially connected to seed regions resulting from the higher threshold. An algorithm in [62] also uses two different thresholds. Initially, a high threshold is applied on the pixel values in order to get rid of cold or barely warm areas, selecting only pixels corresponding to very warm objects. Then, pixels featuring a grey level higher than the lower threshold are selected if they are contiguous to other already selected pixels in a region-growing fashion.

Other types of thresholding solutions are based on locally adjustable thresholds. One of them is the adaptive dual-threshold (ADT) [7] with a local adaptation facility. This algorithm works adaptively under various lighting conditions and contrast levels. A decision threshold is calculated for individual pixels with the knowledge of their neighborhood.

In the FIR systems, the intensity of the pedestrian additionally depends on the clothes, their thickness, and their texture. Thus the objects typically are not homogeneous. To make the pedestrian body as uniform as possible, morphological operations should be used with thresholding for distortion compensation. The dimensions of the structuring elements of morphological operation must be adapted to image resolution.

Other image segmentation methods

In [63], grey-level symmetry, edge symmetry, and edge density are used and analyzed to improve the segmentation process. A candidate generation method driven by the search of the pedestrian head is considered in [64]. Detected ROIs are then resized based on the distance to the camera and then filtered by its vertical edges symmetry. In [65], the ROI is extracted based on discrete key points computed from the phase coherence image using the maximum and minimum moment of covariance.

In [66], the statistical approach for ROI generation is presented. In this solution, a statistical pixel classifier for head detection is used.

The adaptive fuzzy C-means algorithm is employed in the segmentation step in [43]. The method adaptively estimates the required number of clusters and fuses multiple clusters to retrieve the ROI candidates. The second central moment's ellipse is used to prune the set of candidates utilizing the human posture characteristics.

2.4.2. Otsu method

The Otsu (called Otsu after the inventor's name) [60], [67] is a threshold selection method from a gray-level histogram. The algorithm returns a single intensity threshold that separates pixels into two classes (see Figure 15) or even more.

Let an image be divided into a two classes C_0 (background) and C_1 (foreground). The optimal threshold T^* is obtained by maximizing inter-class variance:

$$T^* = \arg\max_{1 \le T \le L} \{\sigma_b^2(T)\} \tag{2}$$

where L is the number of gray levels in the image and $\sigma_b^2(T)$ represents inter-class variance that is defined as follows:

$$\sigma_b^2 = P_0(T)(\mu_0(T) - \mu)^2 + P_1(T)(\mu_1(T) - \mu)^2 \tag{3}$$

where μ represents the mean level of the image, μ_0 represents the mean level of class C_0, μ_1 represents the mean level of class C_1, $P_0(T)$ and $P_1(T)$ denote the cumulative probabilities:

$$P_0(T) = \sum_{i=1}^{T} p_i \tag{4}$$

$$P_1(T) = \sum_{i=T+1}^{T} p_i \tag{5}$$

and

$$\mu_0 = \sum_{i=1}^{T} \frac{i p_i}{P_0(T)} \tag{6}$$

$$\mu_1 = \sum_{i=T+1}^{T} \frac{ip_i}{P_1(T)} \tag{7}$$

Finally, the T^* threshold could be used to perform binarization of the image. The Otsu technique belongs to the global thresholding techniques. Therefore the same threshold value is used for each pixel.

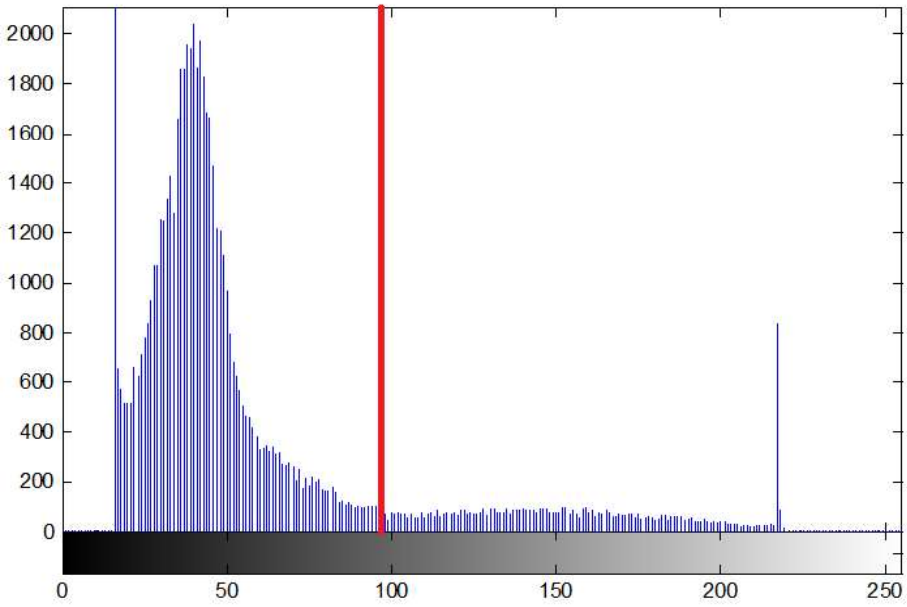


Figure 15. Illustrative histogram extracted from Fig. 2 with marked Otsu's threshold

2.4.3. Locally adaptive dual-threshold

The locally adaptive dual-threshold technique (ADT) initially presented in [7] (modified version presented in [68]) is a variant of locally adaptive thresholding with two thresholds: $T_1(i,j)$ – lower threshold, $T_2(i,j)$ – upper threshold:

$$T_1(i,j) = \frac{1}{2w_{\mathrm{h}}+1} \sum_{m=i-w_{\mathrm{h}}}^{m=i+w_{\mathrm{h}}} I(m,j) \tag{8}$$

$$T_2(i,j) = T_1(i,j) + \lambda \cdot \delta(i,j) \tag{9}$$

where w_{h} is horizontal scanning width (as presented in Figure 16), $I(i,j)$ is a gray-level input image, δ is a standard deviation of the neighboring pixels, and λ is the weight.

In order to produce uniform areas with clear edges, the thresholding algorithm passes through neighboring pixels with values close to the threshold. A value of the upper threshold $T_2(i,j)$ is defined as a sum of the lower threshold and the standard deviation δ of the surrounding pixels:

$$\delta(i,j) = \sqrt{\frac{1}{2w_{\mathrm{h}}+1}\sum_{m=i-w_{\mathrm{h}}}^{m=i+w_{\mathrm{h}}}(I(m,j)-\mu_{\mathrm{h}})^2}, \tag{10}$$

where: w_{h} is a scanning width and μ_{h} is the mean value of the horizontal neighborhood. To control the impact of the standard deviation on the upper threshold the weight λ is added.

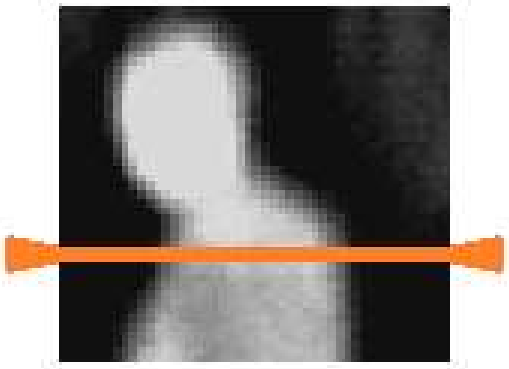


Figure 16. The illustrative figure of a pedestrian and the horizontal scanning line

Finally, the segmentation process is defined as follows:

$$S(i,j) = \begin{cases} 0, & if\ I(i,j) < T_1\ or\ I(i,j) \in (T_1,T_2) \cap I(i-1,j) = 0 \\ 1, & if\ I(i,j) > T_2\ or\ I(i,j) \in (T_1,T_2)\ \cap I(i-1,j) = 1 \end{cases}, \tag{11}$$

where $S(i,j)$ is the segmented binary image after thresholding. For the pixel values greater than T_2 or less than T_1 (arguments i and j were omitted for simplicity) values 1 and 0 are assigned, respectively. If the pixel value is in the range (T_1, T_2) the output value depends on the previous sample in line $S(i-1,j)$.

The algorithm translates the input gray scale image to the binary image, while white objects are the potential candidates to be detected as pedestrians and the background is black.

Examples of thresholding with the ADT method are presented in Figure 17 and Figure 18. Figure 17 shows the image before and after thresholding using the ADT procedure. The red plot with diamond-shaped markers in Figure 18 indicates the value of the intensity (image brightness, temperature of real objects) of one line in the analyzed frame. The blue and green plots show the lower and higher thresholds, respectively. It can be seen that both thresholds adjust to the intensity value but with a much lower frequency (similar to low-pass filtering).

The ADT procedure with horizontal scan lines and its local adaptation separates objects in the horizontal direction (as shown in Figure 17).

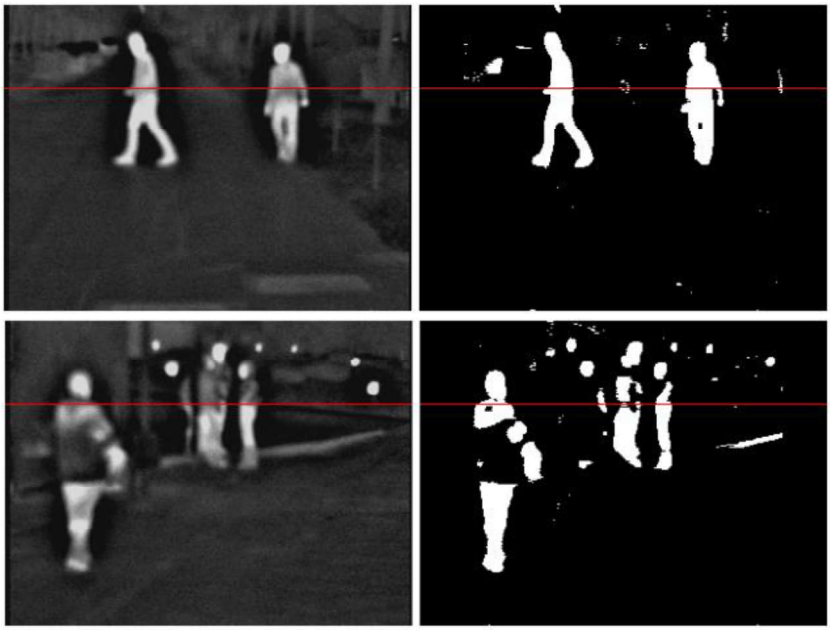


Figure 17. Thermal image before (left) and after (right) ADT segmentation with marked red line that indicates the analyzed line in the graph presented in Figure 18

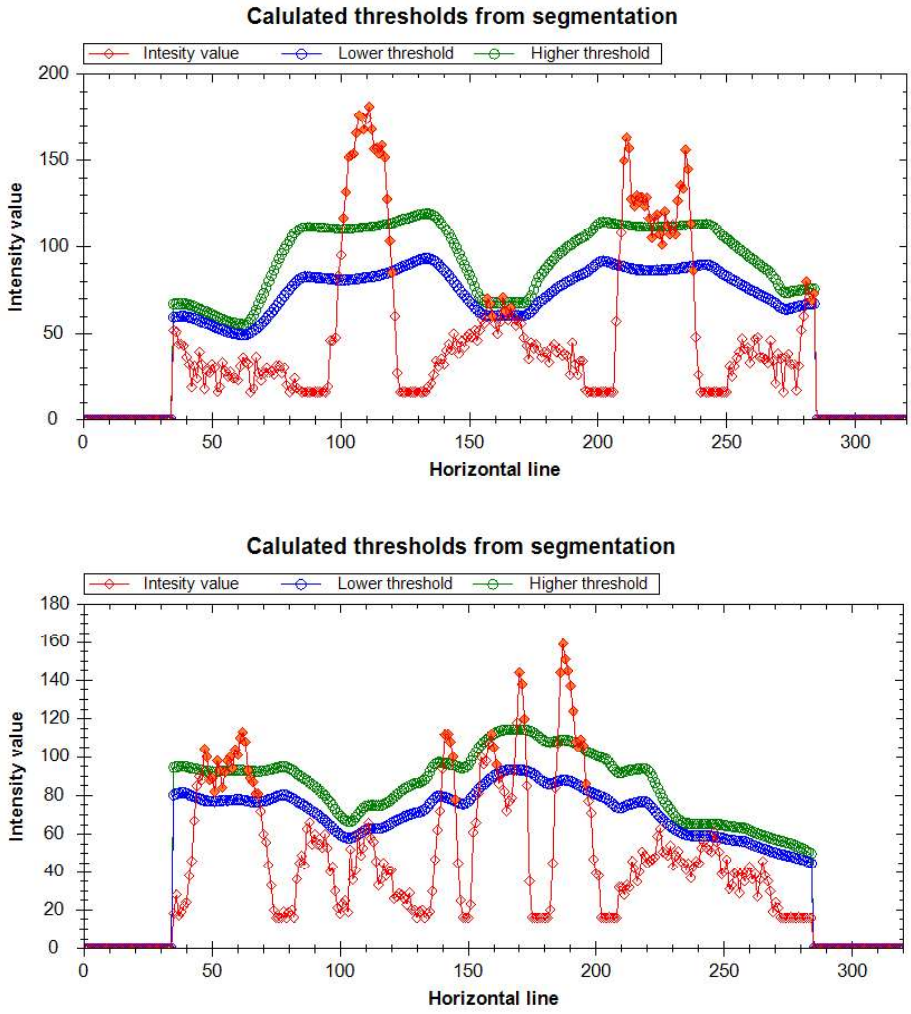


Figure 18. Calculated local thresholds with the ADT procedure for one horizontal line in two frames shown in Figure 17

2.5. Feature extraction

The object classification stage consists of two steps (see Figure 14): feature extraction and final validation with the selected classifier. The feature extraction step brings the most valuable features and reduces the amount of data that describes the object. In a validation step, the classifier finally decides which objects are pedestrians and which are not.

There are many efficient feature extractors used for detection of pedestrians, starting with the basic handcrafted features like histograms of oriented gradients [69], local binary patterns [70], shape context [71], 1D/2D Haar descriptors [72], to plenty of their modifications [44], [71], [73], [74].

Recently, several efficient variants of the HOG were proposed: integral channel features (ICF), for which the HOG descriptors are used together with luminance and UV chrominance components (LUV) [75], the ACF [76] combining HOG channel feature with the normalized gradient magnitude and LUV color channels, and the Checkerboards [77], which are modifications of the ICF. They perform filtering of the HOG+LUV feature channels. The listed above feature extractors have become the state-of-the-art approaches for night-vision pedestrian detection [32].

Contrary to the mentioned handcrafted features, CNNs are now very strongly developed and widely used. The most important CNN models are: AlexNet/CaffeNet [78], [79], VGG [80], ResNet [81]. They allow for self-learning of features and perform significantly better than other approaches. On the other hand, due to their complex structure, they need powerful hardware like e.g. GPUs for real-time computations otherwise operate much slower.

2.5.1. Histogram of oriented gradients

The HOG feature extraction technique [69] was an important step in the development of handcrafted features. Nowadays, it is often used as a part of more advanced solutions, e.g., in the ACF or in the Checkerboards [76], [77]. This method calculates gradients and forms histograms of the gradients orientation. To improve the reliability of the HOG, a local normalization is performed. Finally, the ROI is represented by a locally normalized feature vector constructed from the histograms of orientation.

The first step of this algorithm consists of the calculation of gradients G_i and G_j in the horizontal and vertical axes, respectively, with i and j treated for a moment as continuous variables

$$\nabla I(i,j) = \begin{pmatrix} G_i \\ G_j \end{pmatrix} = \begin{pmatrix} \frac{\partial I}{\partial i} \\ \frac{\partial I}{\partial j} \end{pmatrix} \tag{12}$$

where

$$\frac{\partial I}{\partial i} \approx \frac{I(i+\Delta i, j) - I(i-\Delta i, j)}{2\Delta i} \tag{13}$$

$$\frac{\partial I}{\partial j} \approx \frac{I(i, j + \Delta j) - I(i, j - \Delta j)}{2\Delta j} \tag{14}$$

$$\Delta i, \Delta j = 1 \tag{15}$$

The above formula is equivalent with a convolution operation on the image with filter kernels $\frac{1}{2}[-1\ 0\ 1]$ and $\frac{1}{2}[-1\ 0\ 1]^{\mathrm{T}}$ (but the factor $\frac{1}{2}$ can be omitted). After the gradients are computed, the magnitude and orientation of gradients can be obtained respectively as

$$|\nabla I| = \sqrt{G_i^2 + G_j^2} \tag{16}$$

$$\theta = \arctan\left(\frac{G_i}{G_j}\right) \tag{17}$$

The next step groups the pixels into cells (Figure 19, left-hand, green lattice), which usually have a square shape. For such cells, the orientation histogram (Figure 19, right-hand side) is created using orientation and magnitude. The histogram is divided into nine bins ranging from 0 to 360 degrees or 0 to 180 degrees (the authors claim that for nine bins, the algorithm works the best). Thus, for each pixel in the cell, based on its gradient orientation θ, the magnitude $|\nabla I|$ is proportionally divided between two adjacent bins of the histogram.

After the histograms are calculated, the four adjacent cells are grouped and create a block (Figure 19, left-hand, red rectangle). In this block, a non-normalized vector $\mathbf{v}$ is created, which contains all histograms in a given block (here in four cells).

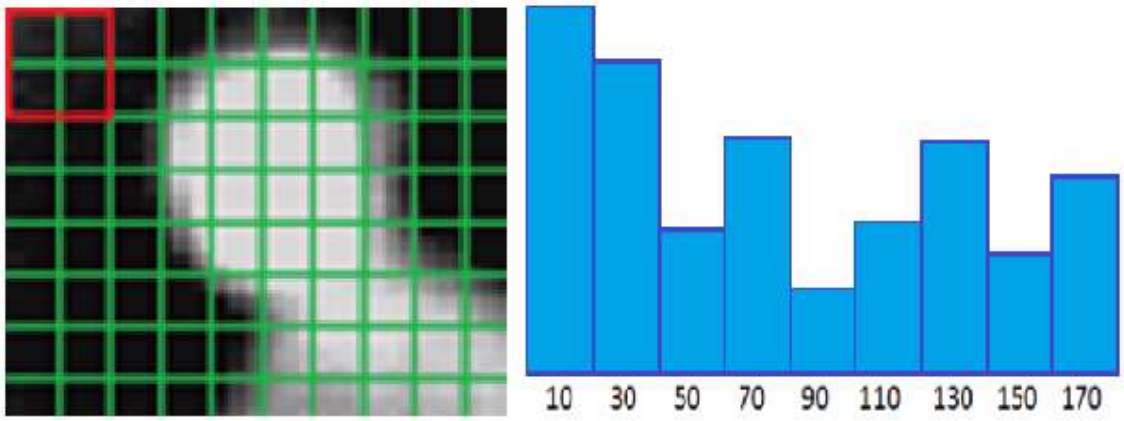


Figure 19. The illustrative image divided into cells and blocks (left) and a histogram of orientation (right)

Therefore, the vector $\mathbf{v}$ is locally normalized in blocks to get $\mathbf{v}_{\mathrm{n}}$ with a formula

$$\mathbf{v}_{\mathrm{n}} = \frac{\mathbf{v}}{\sqrt{\|\mathbf{v}\|_2^2 + e^2}} \tag{18}$$

where e is a small constant. Finally, after the normalization, all vectors are combined into a single feature vector $\mathbf{v}_\mathrm{f}$:

$$\begin{gathered}\mathbf{v}_\mathrm{f} = [\mathbf{v}_{\mathrm{n}[1\ 1]}, \mathbf{v}_{\mathrm{n}[1\ 2]}, \cdots\ \mathbf{v}_{\mathrm{n}[1\ (m-1)]},\ \mathbf{v}_{\mathrm{n}[1\ m]}, \cdots \\ \cdots \mathbf{v}_{\mathrm{n}[2\ 1]}, \mathbf{v}_{\mathrm{n}[2\ 2]}, \cdots\ \mathbf{v}_{\mathrm{n}[2\ (m-1)]}, \mathbf{v}_{\mathrm{n}[2\ m]}, \cdots \\ \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \cdots \\ \cdots \mathbf{v}_{\mathrm{n}[(l-1)1]},\ \mathbf{v}_{\mathrm{n}[(l-1)\ 2]}, \cdots\ \mathbf{v}_{\mathrm{n}[(l-1)\ (m-1)]}, \mathbf{v}_{\mathrm{n}[(l-1)\ m]}, \cdots \\ \cdots \mathbf{v}_{\mathrm{n}[l\ 1]}, \mathbf{v}_{\mathrm{n}[l\ 2]}, \cdots\ \mathbf{v}_{\mathrm{n}[l\ (m-1)]}, \mathbf{v}_{\mathrm{n}[l\ m]}\ ]\end{gathered} \tag{19}$$

where l, m are numbers of blocks on the image in the vertical and horizontal orientation respectively.

2.6. Validation

As it was mentioned in the introduction, a typical approach to the validation process is the use of a single classifier with fixed window size. As a result, all pedestrian candidates must be resized to the classifier resolution before the validation process can start [7], [25], [43], [82].

The most popular classifiers are SVM, AdaBoost (used in the ACF detector), neural networks (including matrices of neurons, self-organizing maps [83], deep CNNs), and various combinations of them. In paper [84], the AdaBoost classifier is used for initial selection while the SVM classifier for the final verification. In [85], the combination of the AdaBoost with the random vector functional link neural network was proposed. In another work [86] a parallel connection of various classifiers was proposed, trained in a complementary manner to each other. The result was high accuracy but low speed.

Three-branch structured SVM classifier based on HIK (histogram intersection kernel) was proposed in [7]. As a result, they achieved increased performance of detection for various heights of pedestrians. The author of this dissertation deeply investigated this technique [7] and expanded it to a form of multi-branch classifiers [87]. Different approach is presented in [88], where a combined classifier was trained for various pedestrian poses composed of four independent AdaBoost classifiers.

The classification part in the CNN models can be realized with fully-connected layers. However, others solutions are also used. In paper [39], CNN together with the AdaBoost classifier is used. The method presented in [13] utilized the deep neural network for classification purposes using multispectral information.

2.7. Summary

In general, most modern approaches to night-vision FIR pedestrian detection are designed similarly to those used for the standard color images. Therefore these techniques achieve similar computational efficiency. However, in the case of pedestrian detection with moving camera, i.e., autonomous vehicles, it is much better to achieve real-time performance without the need of using costly and highly energy-consuming equipment, like e.g., GPGPU.

At night conditions, it is potentially possible to use segmentation by thresholding. This approach allows for a significant acceleration of the entire pedestrian detection process by reducing the ROI area in the image. It uses properties of the FIR spectrum, mainly in the night-time the pedestrians are warmer, therefore brighter than the surroundings. Several techniques for the segmentation of thermal images based on thresholding have been proposed [7], [21], [22], but these techniques do not offer the state-of-the-art accuracy.

The simple assumption that pedestrians are warmer than the surrounding at night is not always valid. Many problems arise with thermal images during segmentation, i.e., the uneven level of the observed temperature of one pedestrian, as well as the temporary loss of thermal contrast between the pedestrian and the surroundings. All the above-mentioned problems should be compensated to avoid the situation that a pedestrian is missed in ROI and, in consequence, not being detected at the segmentation stage. Therefore, there is a need to develop a highly efficient FIR image segmentation algorithm that can offer high accuracy and can compensate common problems that are associated with the thresholding of thermal images.

Regardless of the ROI generation technique used, the quality of the prepared ROIs is very important and significantly affects the effectiveness of the object classification stage. All the advanced segmentation techniques, besides the simplest one, i.e., the sliding window technique, match the ROIs to the outer pedestrian edges in the image. Inaccurate matching the edges of ROI to the outer edges of the pedestrian may lead to ROI covering less than a whole pedestrian. Such too small ROIs may be rejected by the classifier. This will finally increase the number of falsely negative results.

As mentioned in the Introduction, the selection and tuning of the object classification technique also have a significant impact on overall pedestrian detection efficiency. For classification purposes (not only in the context of pedestrian detection), the baseline approach is the use of a single classifier with a fixed input resolution [11], [25], [43]. In the simplest case, to detect pedestrians of various sizes with a single, fixed-size classifier, the scanning window is scaled and shifted through an image. As a result, all pedestrian candidates must be resized (upscaled or downscaled) to the classifier resolution.

The classifiers are often used without an adaptation of the input resolution to the resolution of the specific dataset or camera, which unfortunately is a common practice, especially in the solutions with a complicated structure of the classifier. An example of a very complicated structure of the pedestrian detector is a deep convolutional neural network (CNN) [24], [25], [39], [43], [89]. In the case of CNN, any change in the

resolution of the CNN input layer causes the necessity of adaptation in the other layers. It is quite complicated, and therefore designers try to omit it. For example, in the proposed deep CNN by Kim et al. [24], various grayscale pedestrian images were resized (mainly upscaled, as the smallest pedestrians had 50 pixels in height only) and artificially colorized (!) to fit the input size of the typical, pre-trained model of the CNN detector, which required 224×224 pixels and the color RGB input image format. Such solutions, although simple in implementation, are greatly ineffective.

In this context, it should be emphasized that the high-resolution classifiers often, but not always, offer slightly higher detection performance but always impose additional computational overhead. Formally increasing the pedestrian candidates' resolution does not increase the information content and may have a negative effect during the classifier training. As a result, this can reduce efficiency, especially when there are largely disproportionate pedestrian samples in the training set. Finally, in order to achieve more efficient and faster solutions, there is a need to design an appropriate procedure for tuning of the object classification stage parameters such as input resolution of the classifier.

3. ROI generation procedure for night-vision FIR images

This chapter presents a technique of ROI generation for night-vision FIR images. The first section of this chapter presents the architecture of the proposed solution. The following sections present: the technique of double and triple thresholding, the technique of regions enlargement, which significantly increases the accuracy of the segmentation process, and a set of proposed techniques for filtering ROI candidates for their quick initial selection. Then, the calibration process of the proposed ROI generation procedure is presented with experiments conducted on CVC-14 and KAIST datasets: the selection of values of thresholds and the selection of parameters of the candidate selection process. The summary is presented in the last section of this chapter.

3.1. Algorithm architecture

The algorithm of ROI generation, which is proposed by the author of this dissertation is dedicated to infrared images at night and is based on the assumption that pedestrians are usually brighter than their surroundings. The double and triple thresholding techniques are used for image segmentation to ensure local adaptation in the image. Thanks to the technique of regions enlargement and dividing wide objects, it is possible to significantly increase pedestrian detection accuracy. In order to speed up the ROI generation process and the entire pedestrian detection process (by reducing the number of ROIs per image frame), a set of candidates selection techniques has been proposed. A detailed diagram of the presented approach is shown in Figure 20.

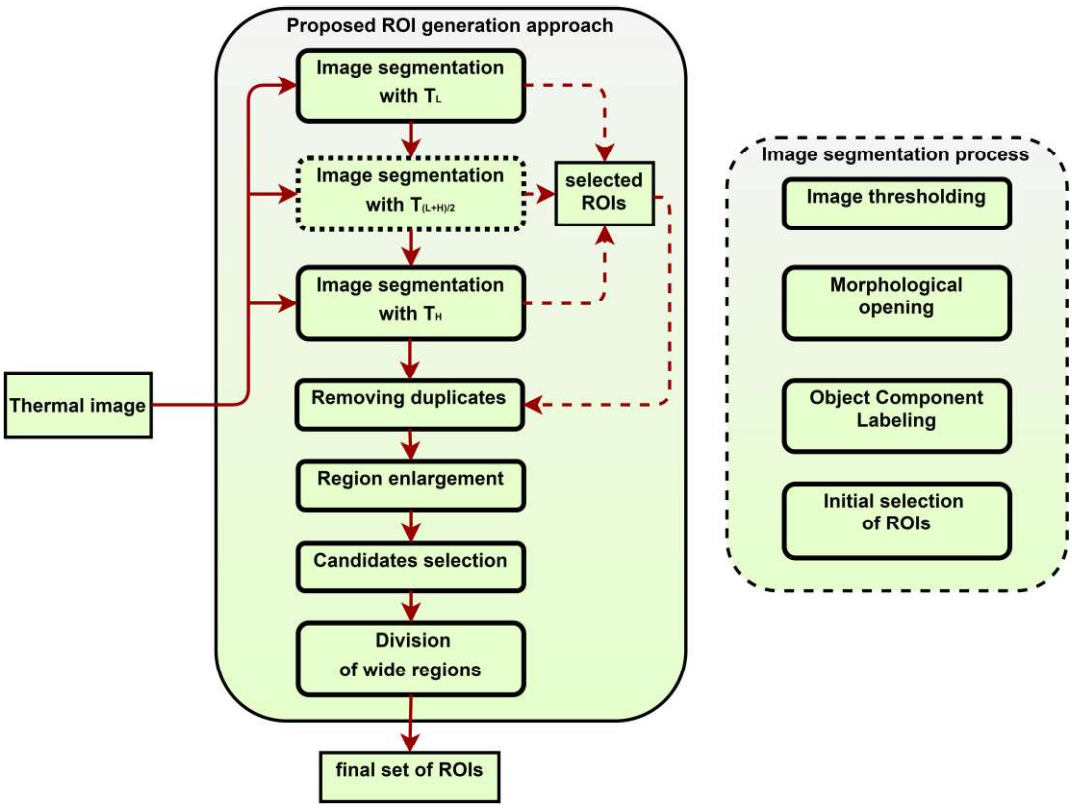


Figure 20. Diagram of the proposed ROI generation approach for thermal images at night

In the procedure, the thresholding process is performed two or three times (depending on the selected version) with various threshold values (see Section 3.2). This process should first be done with a lower threshold and then with a higher one. After each thresholding, an opening operation is performed to remove the smallest objects. Then, objects that constitute inseparable areas in the image are detected. Objects are then pre-selected to speed up the algorithm (described in Section 3.5).

After image segmentation, all ROIs are then summed up, and then duplicates are removed from the set, which could be created by multiple thresholding of the same image (see Section 3.4). Then the set of ROIs is extended with the new additional areas obtained by the regions enlargement technique. It creates additional ROIs by joining all pairs of regions with the same horizontal coordinates (this technique is described in detail in Section 3.3). As a result, the set of ROIs is significantly expanded.

The resulting set of ROIs for a given image is filtered at the candidates selection step using several techniques and parameters (described in Section 3.5). In order to improve accuracy and detect groups of pedestrians, the wide regions (with a low height-to-width ratio) are divided into smaller ROIs (as presented in Section 3.6).

3.2. Double and triple thresholding procedure

As mentioned in the introduction, the thresholding techniques can be divided into the global techniques with one fixed threshold or locally adaptive techniques, where the threshold is calculated separately for each pixel independently.

Several important problems are related to the thresholding process (some of them can be seen in Figure 21). The most important are:

- splitting of objects into many smaller objects, mainly due to uneven infrared radiation of clothed humans,
- connecting two or more pedestrians into one object, or stitching various objects (e.g., pedestrians with non-pedestrians),
- variance in ambient temperature (the camera often automatically adjusts the dynamic range, i.e., the minimum and the maximum values) to obtain the full-scale image,

One fixed, global threshold, in most of the cases cases, is not enough to reach the assumed pedestrian detection accuracy. It is because for a different part of the image, also different thresholds should be used. The Otsu technique [60], which gives some global adaptability, still does not offer local adaptation. To overcome this problem, a local adapting threshold technique was proposed in [7]. However, the threshold is calculated for each pixel, and in consequence, the algorithm performs slowly.

To achieve both: high accuracy and high computational efficiency of the image segmentation, the author proposes a technique of multiple (double or triple) image thresholding with Otsu-based global thresholds.

It is proposed to process the image twice (or three times with additional T_{M} threshold): once with T_{L} global threshold and then with the T_{H} global threshold (as presented in Figure 20). The thresholds should be calculated in the following manner:

$$T_{\mathrm{L}} = T_{\mathrm{Otsu}} + \alpha_{\mathrm{caf}} - \beta, \tag{20}$$

$$T_{\mathrm{M}} = (T_{\mathrm{L}} + T_{\mathrm{H}})/2, \tag{21}$$

$$T_{\mathrm{H}} = T_{\mathrm{Otsu}} + \alpha_{\mathrm{caf}} + \beta, \tag{22}$$

where: T_{Otsu} is Otsu threshold, α_{caf} – is a constant adjusting factor, β – is a difference factor.

The Otsu technique is used to find a baseline threshold to adapt to changes in image dynamics. Based on this threshold value, the thresholds T_{L} and T_{H} are then calculated with two additional factors α_{caf} and β (their values are adjusted to the camera type individually, as presented in section 3.8).

As a result, the proposed technique is a hybrid thresholding technique. Image binarization is performed using one global threshold for all pixels, but it is performed multiple times with different threshold values.

With this technique, thermal contrast is preserved in different parts of the image (see Figure 21), allowing for more accurate pedestrian detection while maintaining high computational efficiency.

Figure 21. Illustrative examples of results of thresholding with lower (middle column) and higher threshold (right column), original images (left column) were taken from the CVC-14 dataset

3.3. Regions enlargement

A new regions enlargement technique is proposed to compensate problem with the splitting of objects into many smaller objects, mainly due to uneven infrared radiation of clothed humans (as can be seen in Figure 21). Background objects, such as buildings, cars, animals, and lights, can be close to the temperature of pedestrians. The desirable difference in magnitude of thermal energy between pedestrians and background objects is also affected by the weather. In low ambient temperature, pedestrians typically wear warmer clothes. This leads to lowering the measured thermal energy by the camera.

Additionally, it increases the variance of the thermal magnitude of a single pedestrian. Parts of the bodies of pedestrians (Figure 22 and Figure 23a, especially heads, arms, and legs) can have much higher intensities (due to relatively high body temperature) than the rest of the bodies covered by cloths with a relatively cold surface. Using typical segmentation methods may result in splitting the pedestrian bodies into parts, putting them to separate samples (see Figure 23b and red rectangles in Figure 23c and Figure 23d).

Taking into account that the pedestrians to be detected typically have vertical postures (although exceptionally in abnormal situations, they may also have horizontal postures), it is proposed to enlarge the number of the analyzed samples by additional samples composed of all possible pairs of original samples aligned vertically. By this means, for all pairs of samples, e.g., a pair $s_1(x_1, y_1, w_1, h_1)$ and $s_2(x_2, y_2, w_2, h_2)$, which have a common part in the horizontal coordinate axis a new merged sample $s_3(x_3, y_3, w_3, h_3)$ is created, which covers the area of both samples and the area between them (where x_i, y_i are image coordinates taken from the upper-left corner). Assuming that x_i, y_i are the smallest coordinate values and w_i, h_i are the width and height, respectively of the sample s_i, $i = 1, 2$, for e.g. $x_2 > x_1$ and $y_2 > y_1$, $w_3 = x_2 - x_1 + w_2$, and $h_3 = y_2 - y_1 + h_2$. An example of vertical alignment is shown in Figure 22. Notice that yellow and green rectangles assign original samples, while the red rectangles correspond to the new merged samples.

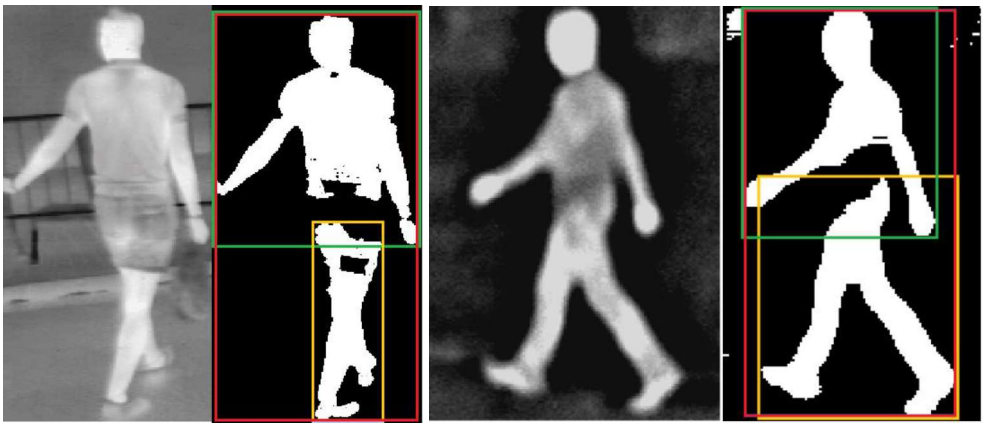


Figure 22. Two illustrate examples of regions enlargement idea: input frame (left), binary image after thresholding (right) with marked component (green and yellow) and enlarged regions (red)

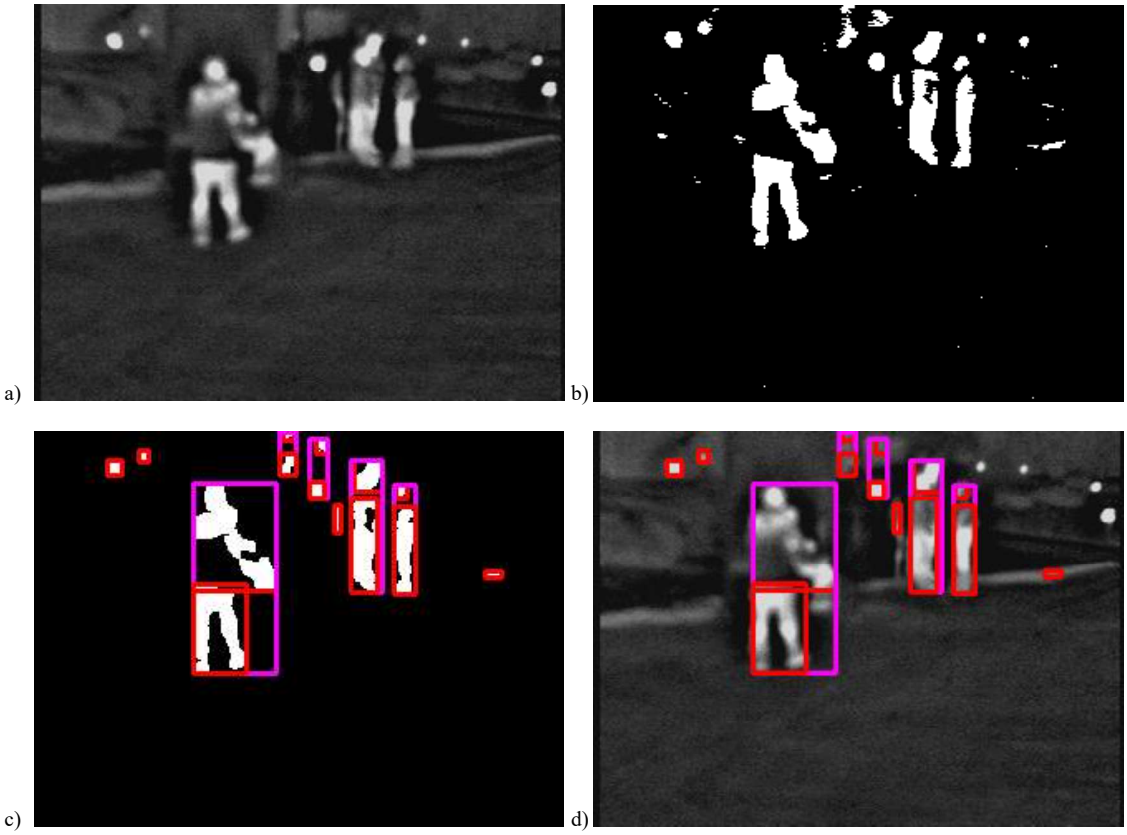


Figure 23. Thresholding-based image segmentation with regions enlargement: (a) input frame, (b) binary image after thresholding, (c) and (d) segments (after thresholding– red rectangles, after regions enlargement – violet rectangles)

3.4. Duplicate detection

After image segmentation with different thresholds, all detected ROIs are then summed. Therefore, duplicate objects may appear in the ROI set.

To remove duplicates from the summed set of ROIs, all ROIs extracted after the second and third thresholding are compared to the ROIs obtained after the first thresholding (with the lowest threshold). For an object (obj1) to be considered as a duplicate of another object (obj2), the following conditions must be met:

$$\frac{A_{\text{obj1}\cap\text{obj2}}}{A_{\text{obj2}}} > \alpha_{\text{sim}}, \tag{23}$$

and

$$\alpha_{\text{sim}} < \frac{A_{\text{obj1}}}{A_{\text{obj2}}} < 2 - \alpha_{\text{sim}} \tag{24}$$

where A_{obj1} and A_{obj2} are the areas of the compared objects (in pixels), $A_{\text{obj1}\cap\text{obj2}}$ is the area of the intersection of these objects, a α_{sim} is the similarity coefficient.

This technique is used before the regions enlargement but is also considered as a candidates selection technique because it reduces the number of ROIs.

3.5. Candidates selection

To speed up the pedestrian detection process, a selection of the obtained ROIs is performed. The use of proposed techniques of coarse selection of pedestrian candidates can significantly reduce their number, having a minor impact on the pedestrian detection accuracy of the segmentation process. At the same time, it is possible to increase the precision of the object classification step by the fact that with a smaller set of ROIs, the classifier will less frequently make erroneous, false-positive detection.

A set of techniques is proposed that is adapted to infrared images. The candidate selection process is divided into two steps. Pre-selection of candidates is carried out immediately after an image segmentation to eliminate a large number of the smallest and flattest objects. This includes three filtering techniques: selection by a minimum area of the object, minimum height-to-width ratio filtering, skew objects filtering.

The main candidates selection step is performed after the regions enlargement and includes the following techniques: minimum and maximum height-to-width ratio, selection by minimum height in relation to the object's position in the image (perspective filtering), selection by minimum object area, and homogeneous areas filtering.

The simplest selection techniques are performed first and the more computationally expensive last to optimize the effectiveness of the candidate selection process. To properly adjust these selection techniques, it is necessary to perform the calibration process.

3.5.1. Height-to-width ratio filtering

Standing or walking pedestrians appear mostly as vertical regions in the image. According to this, it is not necessary to accept ROIs, which are not vertical. In this case, the candidates are filtered by the object aspect ratio α_{HW} (height to width ratio). For pedestrians, this ratio is in the range of 1:1.3 to 1:4 according to their actual distribution in a given dataset.

Despite the presented constant distribution, there is a need to calibrate parameters: minimum $(\alpha_{\mathrm{HW}_{\min}})$ and maximum height-to-width ratio $(\alpha_{\mathrm{HW}_{\max}})$ for each segmentation technique. The detected bounding box of ROIs does not always accurately reflect the pedestrian area in the image. In addition, detection of groups of pedestrians is also performed in the proposed ROI generation technique. Therefore, lower values of the height-to-width ratio are also accepted (see an explanation in Section 3.6).

3.5.2. Perspective filtering

In the discussed applications, the camera is mounted in the front of the vehicle. Thus constraints relating to this perspective can be added.

There is no need to analyze very small objects near the camera. Pedestrians far away from the camera appear smaller in the image. However, their vertical position in the image depends on the distance from the camera, their height, focal length, and the angle at which the camera is set. To avoid a need to set all these parameters, it is proposed to

roughly limit the minimum possible height in relation to the object's position in the image with the formula:

$$h_{\min} = \alpha_{\mathrm{h}} \cdot (y + h) \tag{25}$$

where: (x, y) are coordinates of the upper left corner of a candidate in the image, h – is the height of the object (in pixels), α_{h} - is the height coefficient.

As a result, the value of α_{h} should be selected at the calibration stage of the proposed ROI generation algorithm.

3.5.3. Homogeneous regions filtering

Some of the ROIs can be easily removed due to their homogeneous appearance in the image. Such regions often exist as a part of wide objects, not related to pedestrians. Moreover, the thermal contrast between pedestrians and surroundings is usually high at night. Thus, it is proposed to calculate a standard deviation of ROI (in the gray-scale, taken from the original image) and remove some of them with the intensity below the threshold:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=x}^{x+w}\sum_{j=y}^{j+h}(I(i,j) - \mu)^2}, \tag{26}$$

The decision is taken using the formula:

$$\sigma > \alpha_{\sigma} \tag{27}$$

where: N is the total number of pixels in the region, w, h are width and height of the ROI, $I(i,j)$ is the pixel's intensity in a gray-scale image, and μ is the mean value of ROI. The homogenous coefficient α_{σ} needs to be selected at the calibration stage.

3.5.4. Skew objects filtering

In some cases, after the segmentation process, there are objects whose shape significantly differs from the shape of a pedestrian. An example is the curbs in Figure 21. Their temperature is higher than the ambient temperature, and after the segmentation process (even with a higher threshold), the areas of these objects remained in the binary image. In the case of very flat objects, they could be rejected using a height-to-width ratio. However, objects with a skew shape are not removed with this parameter (due to a similar aspect ratio to the pedestrians).

Therefore, to detect skew objects, it is proposed to use scale and translation invariant second-order normalized central moments η_{20}, η_{02} [90], [91], which are obtained according to the following formula:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\left(1+\frac{i+j}{2}\right)}} \tag{28}$$

where μ_{pq} is a central moment calculated with the formula:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p \, (y - \bar{y})^q I(x, y) \tag{29}$$

where $\bar{x}$ and $\bar{y}$ are components of the object centroid $\{\bar{x}, \bar{y}\}$, $I(x, y)$ is the intensity of the binary image pixel with coordinates $\{x, y\}$.

The final decision is taken using:

$$\eta_{20} > \alpha_\eta \wedge \eta_{02} > \alpha_\eta \tag{30}$$

and

$$\frac{\mu_{00}}{w * h} < \alpha_f \tag{31}$$

where α_η is the skew threshold for normalized central moments, α_f is the threshold for fill factor, *w* and *h* is the width and height of the object.

An additional condition related to the fill factor allows only thin objects to be rejected. Its value was assumed to be $\alpha_f = 1/3$. If the above conditions are met, the object is removed from the set of ROIs.

3.6. Division of wide regions

Segmented groups of pedestrians sometimes form a single region in the binary image. This can be seen in Figure 21 (last row), where three pedestrians, after image thresholding with a lower threshold form one object, and after an image thresholding with a higher threshold, two of them still form a single object.

In some cases, the pedestrians are detected and included in the set of ROIs, but they are incorrectly represented as one pedestrian candidate. Such ROI is wider than this with a single pedestrian and therefore tends to have a much lower height-to-width ratio and can be rejected with $\alpha_{\mathrm{HW}_{\min}}$ parameter. In addition, the classifier is usually trained to classify single pedestrians, so classifying an object with several pedestrians may return a negative result. Additionally, the scaling of the wide ROI to the classifier resolution may result in a significant change of the aspect ratio of ROI (e.g., a change from 0.7 to 2) and then in an incorrect classifier decision.

To solve this problem, it is proposed to divide the wide ROIs into smaller regions. The division is made vertically according to the following rules:

- split into two ROIs for α_{HW} in the range of 1:1.8 to 1:1.2,
- split into three ROIs for α_{HW} in the range of 1:1.2 to $\alpha_{\mathrm{HW}_{\min}}$.

The boundary values (1.2 and 1.8) were selected experimentally on the basis of the distribution of ROIs which included two or three pedestrians.

For ROI with a resolution of 90×100 (width×height), 3 additional ROIs with resolutions of 30×100 each are created. In addition, the large ROI is not rejected and remains in the set of ROIs. As a result, the value of the parameter $\alpha_{\mathrm{HW}_{\min}}$ should be smaller than it results from the distribution of the height-to-width ratio of pedestrians.

3.7. The adaptive limiting of the number of ROIs

The number of ROIs for the proposed ROI generation method may increase significantly when a lot of warm objects appear in the infrared image. In this case, the number of ROIs will be quite large, and with the regions enlargement technique applied, it can even increase significantly. As a result, the pedestrian detection process may temporarily slow down because the classifier (at object classification stage) has many more ROIs to process.

To ensure better stability of the pedestrian detection algorithm, it is proposed to optionally limit the maximum number of ROIs per one image frame along with the $l_{\mathrm{ROIs}_{\max}}$ parameter. With this parameter, it is possible to stabilize the computational efficiency of the entire pedestrian detection process, if required. This minimize a variation of the processing time per frame, what is especially important in the real-time solutions.

After the ROI generation procedure, it is checked if the number of detected ROIs is greater than $l_{\mathrm{ROIs}_{\max}}$. If so, the candidates selection process is performed again. However, this time with a changed, more restrictive values of parameters (smaller or larger, depending on type) of candidates selection process by 10% (the full set of changed parameters is presented in Table 6Table 5). As a result, more ROIs are rejected with each step. The procedure is repeated until the required limit of ROIs is reached.

This procedure may reduce the pedestrian detection accuracy, so it should be used carefully with a quite high value of the parameter $l_{\mathrm{ROIs}_{\max}}$ (the impact of this technique on the accuracy of the segmentation process is tested in Section 3.8).

3.8. Algorithm calibration

This section presents the calibration process of the proposed ROI generation procedure for thermal images along with experiments for two datasets: CVC-14 and KAIST (the datasets are described in detail in Section 2.2). These datasets were selected because they offer thermal imaging sequences (with annotated pedestrians) recorded in night conditions from the vehicle.

In the beginning of the experiments, thresholds were selected for each dataset individually. In both cases, the proposed segmentation procedure with double and triple thresholding was compared. Segmentation with one threshold was also added to the comparison. Moreover, to measure the effectiveness of the regions enlargement technique, each time the proposed ROI generation process was performed with and without this technique.

In addition, thresholds calculated based on the Otsu method (as presented in Section 3.2) were also compared to the thresholds with fixed values.

Then, in the next step, the values of the parameters of the candidate selection process were selected for each dataset. All parameters values of the proposed ROI generation process are presented in Table 5.

Table 5. Set of parameters of the proposed ROI generation technique

Type	Name of the parameter	Symbol
Otsu-based thresholds	constant adjusting factor	α_{caf}
	difference factor	β
Initial ROIs selection	minimum object area	A_{init}
	skew threshold	α_{η}
Candidates selection	minimum ROI area	A_{ROI}
	minimum height-to-width ratio	$\alpha_{\text{HW}_{\text{min}}}$
	maximum height-to-width ratio	$\alpha_{\text{HW}_{\text{max}}}$
	similarity coefficient	α_{sim}
	homogeneous coefficient	α_{σ}
	height coefficient	α_{h}
	maximum number of ROIs per one image	$l_{\text{ROIs}_{\text{max}}}$

3.8.1. Methodology for evaluating the results

To present the results of experiments, a standard Caltech methodology for pedestrian detection was adopted from [92]. To measure the accuracy of detecting pedestrians, the miss rate (MR) is used together with false-positives per image (FPPI), the number of selected ROIs per frame (PR), mean calculation time (MCT), and frames per second (FPS) metrics, which are calculated with the following manner:

$$\text{MR} = \frac{falsly\ rejected\ positive\ samples}{number\ of\ positive\ samples} \tag{32}$$

$$\text{PR} = \frac{total\ number\ of\ selected\ ROIs}{number\ of\ tested\ frames} \tag{33}$$

$$\text{FPPI} = \frac{falsly\ accepted\ negative\ samples}{number\ of\ tested\ frames} \tag{34}$$

$$\text{MCT} = \frac{total\ calculation\ time}{number\ of\ tested\ frames} \tag{35}$$

$$\text{FPS} = \frac{number\ of\ tested\ frames}{total\ calculation\ time} \tag{36}$$

where positive samples are those related to pedestrians.

The miss rate efficiency of the tested ROI generation procedure was verified as follows: a single pedestrian must be selected as one window, and the ROI bounding rectangle must cover at least 40% of the pedestrian area (based on dataset annotations). Additionally, it must not be stitched with other objects like other pedestrians, cars, trees, houses, etc. Such low threshold value (40%) is acceptable, because the area of pedestrians annotations (provided by the authors of dataset) is often significantly larger than the real pedestrian area in the image. Moreover, pedestrians sometimes does not fit perfectly into the ROI in total, and ROI could be a bit smaller than the pedestrian area

(but it is still possible to correctly classify such ROI), this problem is discussed with proposed solution in Chapter 4.

3.8.2. Calibration on CVC-14 dataset

The initial values of parameters of candidates selection were selected experimentally, and their values are presented in Table 6. The performed tests required candidates selection techniques from the very beginning because without these techniques, the number of generated ROIs would be very large.

Table 6. Pre-selected parameters values for the CVC-14 dataset

Type	Name of the parameter	Symbol	Initial values
Initial ROIs selection	minimum object area	A_{init}	30 pixels
	skew threshold	α_{η}	0.14
Candidates selection	minimum ROI area	A_{ROI}	250 pixels
	minimum height-to-width ratio	$\alpha_{\text{HW}_{\text{min}}}$	0.9
	maximum height-to-width ratio	$\alpha_{\text{HW}_{\text{max}}}$	6.5
	similarity coefficient	α_{sim}	0.65
	homogeneous coefficient	α_{σ}	28
	height coefficient	α_{h}	0.45
	maximum number of ROIs per one image	$l_{\text{ROIs}_{\text{max}}}$	50

Selection of thresholds

The experiments with the selection of thresholds were performed for the proposed ROI generation procedure.

The results of the single thresholding are shown in Figure 24 and Figure 25, and the results for the double and triple thresholding are presented in Tables 7-14. The summary of the best results for double and triple thresholding is presented in Table 15.

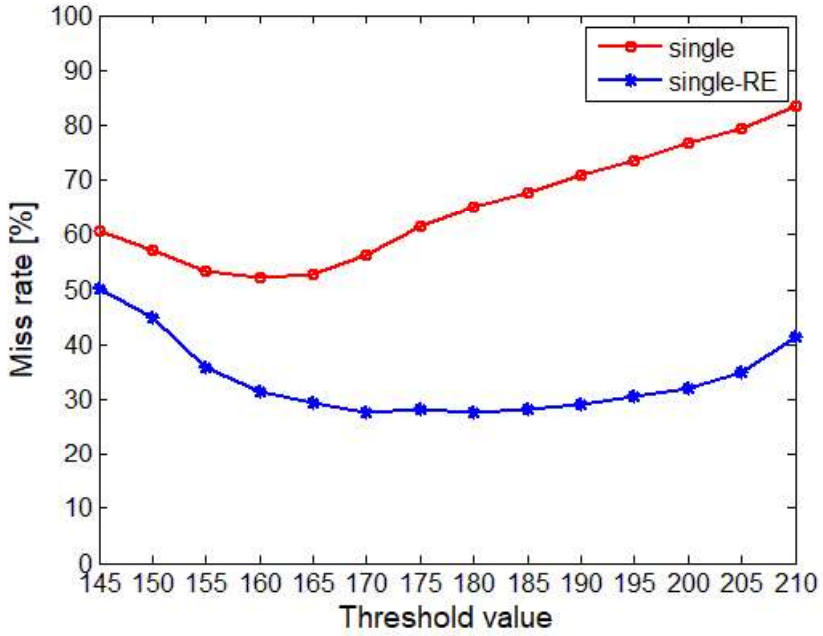


Figure 24. Miss rate for various threshold values for the CVC-14 dataset with a single fixed threshold (in legend: single – denotes segmentation without regions enlargement, single-RE – denotes segmentation with regions enlargement)

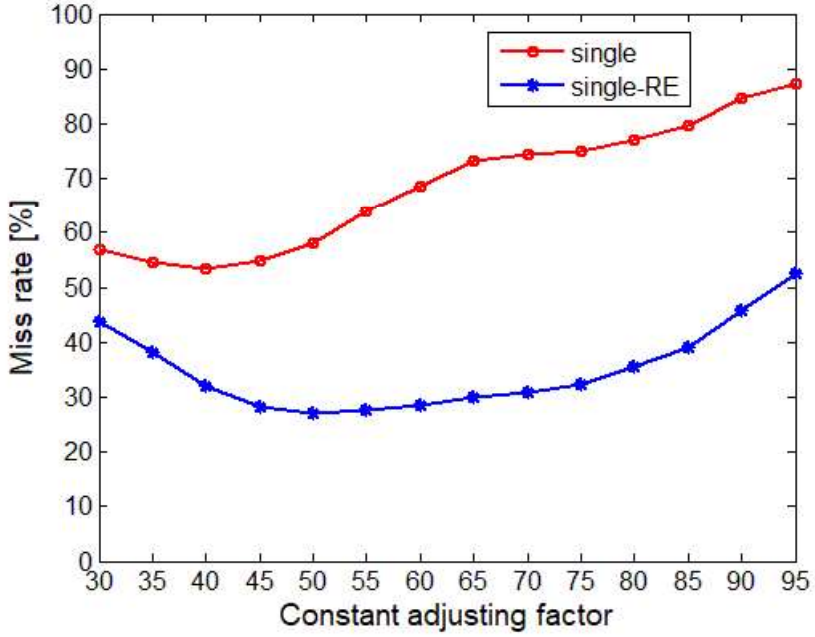


Figure 25. Miss rate for various thresholds for the CVC-14 dataset with a single Otsu-based threshold (in legend: single – denotes segmentation without regions enlargement, single-RE – denotes segmentation with regions enlargement)

Table 7. Miss rate [%] for various thresholds for the CVC-14 dataset with double fixed thresholds and with regions enlargement

T_L \ T_H	190	195	200	205	210	215	220	225	230	235	240	245	250
120	19.4	20.7	21.8	24.0	28.8	33.1	40.7	48.2	56.0	66.8	79.5	89.0	89.0
125	18.0	18.9	19.8	21.2	25.3	29.7	35.8	42.6	50.3	62.5	73.5	82.2	82.2
130	16.6	16.8	17.0	18.5	22.9	26.2	31.8	38.2	45.9	58.0	69.1	76.0	76.0
135	15.3	14.8	15.4	16.1	20.3	23.7	29.0	34.7	41.6	52.6	63.2	69.4	69.4
140	15.1	14.2	14.3	15.2	18.7	21.8	26.6	31.7	38.6	47.4	54.9	60.2	60.2
145	14.6	13.7	13.1	13.8	16.6	18.8	23.0	27.8	33.9	39.7	45.9	50.1	50.1
150	14.6	13.8	12.9	13.1	15.3	17.0	20.5	24.7	30.0	35.9	41.0	45.0	45.0
155	14.5	13.2	12.1	11.9	12.9	13.8	16.3	19.6	24.1	28.3	32.8	35.8	35.8
160	16.1	14.6	13.8	13.2	13.2	13.7	16.3	18.9	22.4	25.9	29.3	31.5	31.5
165	17.6	16.3	15.4	14.7	14.5	15.1	17.3	19.4	22.1	24.8	28.1	29.4	29.4
170	18.9	17.5	16.4	15.3	15.6	15.5	17.0	18.9	21.3	24.5	26.9	27.7	27.7
175	21.1	19.5	18.6	17.3	17.5	17.7	19.2	20.6	22.6	25.6	27.3	28.3	28.3
180	23.0	21.2	20.2	19.2	19.4	19.7	20.4	21.7	23.7	25.2	26.8	27.5	27.5

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 8. Miss rate [%] for various thresholds for the CVC-14 dataset with double fixed thresholds and without regions enlargement

T_L \ T_H	190	195	200	205	210	215	220	225	230	235	240	245	250
120	65.3	68.0	70.9	73.1	76.9	80.2	82.7	85.1	88.1	91.9	92.3	92.3	92.3
125	62.2	64.6	67.1	69.2	72.9	76.2	78.7	80.9	83.5	86.9	87.3	87.3	87.3
130	59.6	61.8	64.1	65.9	69.6	72.9	75.4	77.2	79.5	82.4	82.8	82.8	82.8
135	55.8	58.0	59.7	61.1	64.7	68.0	70.4	72.3	74.1	76.0	76.3	76.3	76.3
140	51.1	53.0	54.5	55.7	59.3	62.4	64.7	66.2	67.5	68.6	68.7	68.7	68.7
145	46.6	48.2	49.5	50.6	53.8	56.9	58.5	59.7	60.4	60.8	60.8	60.8	60.8
150	45.4	46.7	47.8	48.9	51.8	54.6	55.7	56.6	57.2	57.3	57.3	57.3	57.3
155	44.3	45.0	45.9	46.4	48.8	51.1	52.2	52.8	53.4	53.4	53.4	53.4	53.4
160	45.5	46.1	46.6	47.1	48.9	50.2	51.3	51.7	52.1	52.2	52.2	52.2	52.2
165	48.2	48.5	48.9	49.2	50.1	51.5	52.1	52.5	52.8	52.9	52.9	52.9	52.9
170	52.9	52.9	53.1	53.0	53.8	54.8	55.4	55.8	56.1	56.2	56.2	56.2	56.2
175	58.9	58.8	58.8	58.8	59.5	60.3	60.9	61.2	61.5	61.5	61.5	61.5	61.5
180	63.1	62.7	62.7	62.7	63.4	64.2	64.6	64.7	65.0	65.1	65.1	65.1	65.1

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 9. Miss rate [%] for various thresholds for the CVC-14 dataset with double Otsu-based thresholds and with regions enlargement

α_{caf} \ β	8	10	12	14	16	18	20	22	24	26	28	30	32
30	29.0	26.9	24.0	21.4	20.6	18.9	18.7	18.5	18.8	19.1	19.3	19.6	20.2
35	24.7	22.8	21.3	19.9	19.6	18.7	18.9	18.6	19.0	18.7	18.4	19.3	19.9
40	22.3	21.2	20.2	20.2	19.0	19.1	18.8	18.0	18.3	18.1	18.6	18.8	18.0
45	20.5	20.3	20.1	19.7	18.6	17.7	18.0	18.4	16.9	16.9	16.6	17.0	17.6
50	20.5	19.6	18.9	18.6	18.7	18.3	17.0	16.0	15.9	15.5	16.2	16.1	16.8
55	20.7	20.5	19.3	17.7	16.9	15.5	15.1	15.5	15.6	15.1	15.6	16.3	17.7
60	21.6	20.6	18.8	18.3	16.8	16.3	15.0	14.7	14.6	15.4	16.4	17.6	19.1
65	22.3	21.3	20.2	19.3	17.9	16.5	15.8	15.1	14.9	16.0	16.5	18.1	20.7
70	24.1	22.3	20.6	19.7	18.5	17.4	17.0	16.5	17.0	16.5	17.4	18.4	20.5
75	24.5	23.0	21.7	21.1	19.9	20.1	19.4	18.3	18.2	18.4	18.5	19.6	20.5
80	26.0	24.4	23.9	23.9	23.0	22.0	21.5	21.5	20.9	20.7	20.9	20.8	20.7
85	29.1	28.3	26.8	25.9	25.2	25.3	24.6	23.7	23.7	23.4	23.3	23.6	23.0
90	33.5	32.0	31.0	29.6	28.0	27.3	26.6	26.3	26.5	26.6	26.0	26.2	26.1
95	39.7	37.5	35.7	33.6	32.9	30.4	29.9	28.9	28.8	28.5	28.8	28.7	28.0

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 10. Miss rate [%] for various thresholds for the CVC-14 dataset with double Otsu-based thresholds and without regions enlargement

α_{caf} \ β	8	10	12	14	16	18	20	22	24	26	28	30	32
30	46.2	44.1	44.0	43.1	43.9	45.3	47.8	49.9	52.6	56.0	57.3	60.2	63.9
35	44.1	43.0	42.4	43.4	44.3	46.5	49.4	51.7	54.5	57.6	60.7	62.8	63.6
40	44.4	43.2	42.9	44.5	46.0	48.0	50.3	53.4	56.2	57.3	59.2	60.8	62.0
45	45.7	45.4	45.9	45.8	46.7	49.0	50.1	51.7	53.5	54.7	56.1	57.3	59.0
50	48.9	48.0	47.4	47.3	48.1	48.5	47.2	47.8	49.5	50.2	52.9	55.1	56.7
55	53.0	51.5	49.6	48.2	47.4	47.2	46.6	47.1	47.1	47.9	50.1	51.6	55.3
60	57.9	55.1	52.7	51.0	49.2	48.2	47.9	47.0	47.3	48.6	50.2	51.6	53.1
65	63.0	61.0	58.4	55.7	54.0	51.6	50.6	49.8	49.9	50.4	50.8	51.4	52.8
70	68.6	66.1	63.4	62.4	59.6	57.4	55.8	53.8	52.8	52.0	51.7	51.8	51.9
75	71.4	70.6	69.9	67.6	65.2	63.5	61.8	59.6	57.2	56.0	54.4	54.2	53.5
80	73.7	73.0	72.5	71.7	71.1	69.8	67.3	65.0	64.3	61.5	59.3	57.8	55.9
85	74.7	74.5	74.0	73.9	73.5	73.0	72.7	71.8	69.4	67.0	65.4	63.6	61.1
90	77.5	76.7	75.5	75.0	74.9	74.4	74.0	73.9	73.2	72.8	71.1	68.5	66.0
95	81.5	79.4	78.2	77.2	76.4	75.4	75.0	74.6	74.5	74.2	73.8	73.3	72.2

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 11. Miss rate [%] for various thresholds for the CVC-14 dataset with triple fixed thresholds and with regions enlargement

T_L \ T_H	190	195	200	205	210	215	220	225	230	235	240	245	250
120	10.6	9.8	9.2	8.4	8.8	9.0	10.6	12.3	15.2	18.0	18.8	19.7	19.5
125	11.0	10.3	9.2	8.9	8.4	9.1	10.2	11.8	14.0	16.2	17.8	18.3	17.7
130	11.2	9.9	9.1	8.0	8.7	8.7	10.1	11.3	13.1	14.8	16.4	17.1	16.6
135	10.9	9.8	9.2	8.3	8.4	9.0	9.8	10.9	12.7	14.8	15.4	15.3	15.5
140	12.1	10.8	10.3	9.1	9.8	9.9	10.6	11.7	13.4	14.1	14.5	15.4	14.2
145	12.6	11.1	10.1	9.0	9.3	9.7	10.5	11.2	12.4	13.0	14.3	13.7	13.3
150	13.5	12.4	11.2	9.8	10.3	10.6	11.0	11.0	12.2	13.5	13.5	13.0	12.9
155	13.8	12.6	10.8	10.1	10.2	10.5	10.3	10.4	11.7	11.9	12.2	12.1	12.1
160	15.4	13.5	12.7	11.7	11.6	11.9	12.0	12.2	12.7	13.0	13.4	13.5	13.2
165	16.9	15.4	14.4	13.6	13.4	13.4	13.6	13.3	14.1	14.3	14.7	14.7	14.7
170	18.4	16.4	15.4	14.5	14.4	14.7	14.5	14.5	15.0	15.4	15.1	15.4	15.6
175	20.3	19.2	17.4	16.5	16.3	16.5	16.5	16.7	17.2	16.8	17.3	17.5	17.3
180	22.8	20.6	19.2	18.5	18.5	18.6	18.4	18.9	18.7	18.7	19.3	19.1	19.7

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 12. Miss rate [%] for various thresholds for the CVC-14 dataset with triple fixed thresholds and without regions enlargement

T_L \ T_H	190	195	200	205	210	215	220	225	230	235	240	245	250
120	40.8	40.9	42.8	43.0	46.3	48.3	51.2	54.6	56.8	57.7	59.9	61.1	62.3
125	38.9	40.9	41.2	43.5	45.3	48.6	52.3	54.2	55.3	57.6	58.7	59.7	61.3
130	39.4	39.5	41.9	42.8	45.8	49.6	51.7	52.6	55.0	56.1	57.1	58.7	59.6
135	37.1	39.4	40.3	42.3	45.6	48.1	49.4	51.3	52.6	53.5	55.0	55.8	56.5
140	37.8	38.3	40.1	41.8	44.3	45.7	47.5	48.5	49.4	50.9	51.1	51.8	53.0
145	37.2	38.0	39.2	39.9	41.4	43.7	44.4	45.0	46.5	46.6	47.1	48.2	48.5
150	38.8	39.7	40.2	40.3	42.5	44.0	44.1	45.3	45.4	45.8	46.7	47.0	47.8
155	40.9	40.5	40.4	41.1	42.2	43.2	44.0	44.2	44.5	45.0	45.2	45.9	46.0
160	43.0	42.6	42.9	43.4	44.1	45.1	45.2	45.3	46.1	46.1	46.6	46.7	47.1
165	45.9	46.1	46.3	46.4	47.3	47.8	47.9	48.3	48.4	48.9	49.0	49.2	49.8
170	51.7	51.3	51.5	51.6	51.9	52.6	52.6	52.7	53.1	53.1	53.0	53.5	53.8
175	58.5	58.2	58.0	57.5	57.9	58.4	58.3	58.8	58.9	58.8	59.3	59.5	59.9
180	63.0	62.4	61.9	61.7	62.0	62.3	62.7	62.7	62.7	63.2	63.4	63.7	64.2

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 13. Miss rate [%] for various thresholds for the CVC-14 dataset with triple Otsu-based thresholds and with regions enlargement

α_{caf} \ β	8	10	12	14	16	18	20	22	24	26	28	30	32
30	17.5	17.5	17.1	17.0	16.1	15.5	15.6	14.8	14.2	13.2	12.3	11.7	11.6
35	16.7	16.1	15.7	15.0	14.6	14.6	13.7	13.6	12.3	11.6	11.6	11.4	11.4
40	15.0	14.4	13.9	13.0	12.5	11.5	10.7	10.3	10.5	9.9	9.6	10.0	10.4
45	15.2	13.9	13.2	12.2	12.2	11.7	11.6	10.9	9.8	9.4	10.0	9.6	10.8
50	14.0	13.1	13.0	12.7	11.6	10.6	10.6	10.3	10.1	9.7	9.8	10.2	10.8
55	14.4	13.9	12.6	11.7	11.1	10.9	10.5	10.4	10.6	11.0	10.8	11.3	11.3
60	12.9	12.6	12.0	11.7	11.3	11.3	11.3	11.0	11.7	12.0	11.7	12.1	12.5
65	13.7	12.9	12.6	11.8	12.0	12.0	12.5	12.2	12.3	12.8	13.4	13.1	13.7
70	14.8	14.4	13.5	13.0	12.3	12.3	12.1	12.7	13.0	12.8	13.1	14.1	14.2
75	16.5	15.9	15.5	14.8	14.4	13.8	13.2	12.8	12.3	13.2	13.3	13.7	14.0
80	19.1	18.3	17.3	16.7	16.4	15.6	15.5	14.5	14.2	13.7	14.1	14.1	14.2
85	20.8	20.3	19.1	18.9	18.0	16.9	16.5	16.1	15.8	15.6	14.8	14.4	14.8
90	23.7	23.0	22.1	21.1	20.5	19.6	19.0	18.0	17.2	16.8	16.2	16.0	15.1
95	26.7	25.5	24.5	24.2	23.3	22.1	21.4	20.4	20.3	19.4	18.3	17.7	17.2

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 14. Miss rate [%] for various thresholds for the CVC-14 dataset with triple Otsu-based thresholds and without regions enlargement

α_{caf} \ β	8	10	12	14	16	18	20	22	24	26	28	30	32
30	38.8	39.5	40.9	40.8	41.8	43.6	45.1	45.2	45.2	44.3	44.6	44.9	44.8
35	39.5	39.9	40.9	41.6	42.4	42.1	41.8	42.2	42.7	43.2	43.4	43.6	44.4
40	38.8	39.3	39.4	40.1	39.9	40.7	40.8	41.3	41.3	42.4	43.0	44.1	45.1
45	38.8	38.7	38.6	39.0	39.4	39.9	41.2	42.0	42.5	43.4	44.2	44.9	46.0
50	39.2	39.9	39.2	39.9	40.6	41.0	42.0	42.8	43.9	45.5	46.6	47.2	48.2
55	42.2	41.7	41.3	42.1	42.0	43.4	44.6	45.6	46.3	47.5	48.6	50.2	51.4
60	44.2	43.7	44.0	44.3	43.9	44.2	45.8	45.9	47.8	49.0	50.1	51.8	52.7
65	47.7	46.5	46.6	46.0	45.8	46.8	46.8	47.2	48.5	49.4	51.2	53.0	54.2
70	51.3	49.8	48.4	47.7	46.9	46.0	46.2	47.1	47.3	46.9	47.4	49.1	50.1
75	58.1	55.3	53.5	51.3	50.2	48.5	47.5	47.2	46.9	46.5	47.0	46.6	47.2
80	63.2	62.3	59.3	56.9	55.0	52.5	50.9	49.5	48.5	47.8	47.0	47.0	47.5
85	69.4	67.1	64.5	62.8	61.0	58.5	55.7	53.8	51.6	50.6	49.2	48.4	48.3
90	72.4	71.6	71.0	69.3	66.5	64.0	62.9	59.9	57.7	55.8	53.5	52.2	51.0
95	73.8	73.6	73.0	72.1	71.5	70.4	68.1	65.4	63.7	61.8	59.4	56.7	55.3

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 15. Best experimental results (based on lowest MR value) obtained for the CVC-14 dataset with the proposed ROI generation with double and triple thresholding and with region enlargement

		Double thresholding				**Triple thresholding**			
FIXED	T_L	155	155	150	155	130	135	135	120
	T_H	205	200	200	210	205	205	210	205
	MR [%]	11.9	12.1	12.9	12.9	8.0	8.3	8.4	8.4
	PR	19.2	19.1	19.8	18.6	27.1	26.7	29.9	27.7
	MCT(*) [ms]	7	7	7	7	16	15	14	17
OTSU	α_{caf}	60	60	60	65	45	40	45	50
	β	24	20	24	22	40	42	44	40
	MR [%]	14.6	14.7	15	15.1	9.4	9.6	9.6	9.7
	PR	18.1	17.8	17.4	16.8	26.9	28.1	27.5	25.1
	MCT(*) [ms]	7	7	6	6	18	21	18	17

(*) The mean calculation time MCT was calculated for single-core of Intel Core i7-870 CPU

Presented results of the selection of thresholds (Figure 24 and Figure 25 and Tables 7-14) show that the accuracy of the proposed ROI generation procedure with double and triple thresholding is much higher than with the single thresholding. The difference in the achieved MR values equals to 18.7% (between lowest MR values for single and triple thresholding). Therefore, no further analysis for this dataset will be performed for single thresholding.

The regions enlargement technique has also a significant impact on the MR coefficient. Thanks to this technique, it was possible to lower significantly the MR parameter, e.g. with double thresholding and fixed thresholds, the MR value decreased from 46.4% to 11.9% for thresholds values T_L = 155 and T_H = 205 (see Table 7 and Table 8), with triple thresholding and fixed thresholds, the MR value decreased from 42.8% to 8.0% for thresholds values T_L = 130 and T_H = 205 (see Table 11 and Table 12). A similar tendency can be observed in the case of Otsu-based thresholds.

The use of the triple thresholding also reduces MR compared to double thresholding. The difference is approximately 3-5%, e.g., for the triple thresholding with fixed thresholds, the MR decreased from 11.9% to 8.0% compared to the double thresholding, and for the triple thresholding with Otsu-based thresholds, the MR decreased from 14.6% to 9.4% compared to the double thresholding (see Table 15). Moreover, it is also important that the MR value is more stable, less dependent on the threshold values for triple thresholding (see Tables 7-14). On the other hand, the mean processing time MCT for the triple thresholding increases significantly compared to the double thresholding, from approximately 7 ms to 17 ms, and the number of selected ROIs per frame increases from approximately 18 to 27.

The experiments also show that using the Otsu-based thresholds approach is slightly less advantageous than the fixed-based thresholds approach, e.g., the MR value for Otsu-based thresholds is higher for triple thresholds and equals to 9.4% compared to 8.0% for fixed-based thresholds (see Table 15). However, the use of Otsu-based thresholds is a more reliable solution due to the possibility of adapting to changes in image dynamics.

Candidates selection

In the next step, a plenty of experiments were conducted to verify the impact of changes in the values of candidates selection parameters on the values of MR and MCT metrics and to select sets of optimal parameters values for the double and triple thresholding.

The initial values of tested parameters were selected experimentally and their values are presented in Table 16Table 6. The results of the experiments are presented in the graphs in Figure 26 and Figure 27.

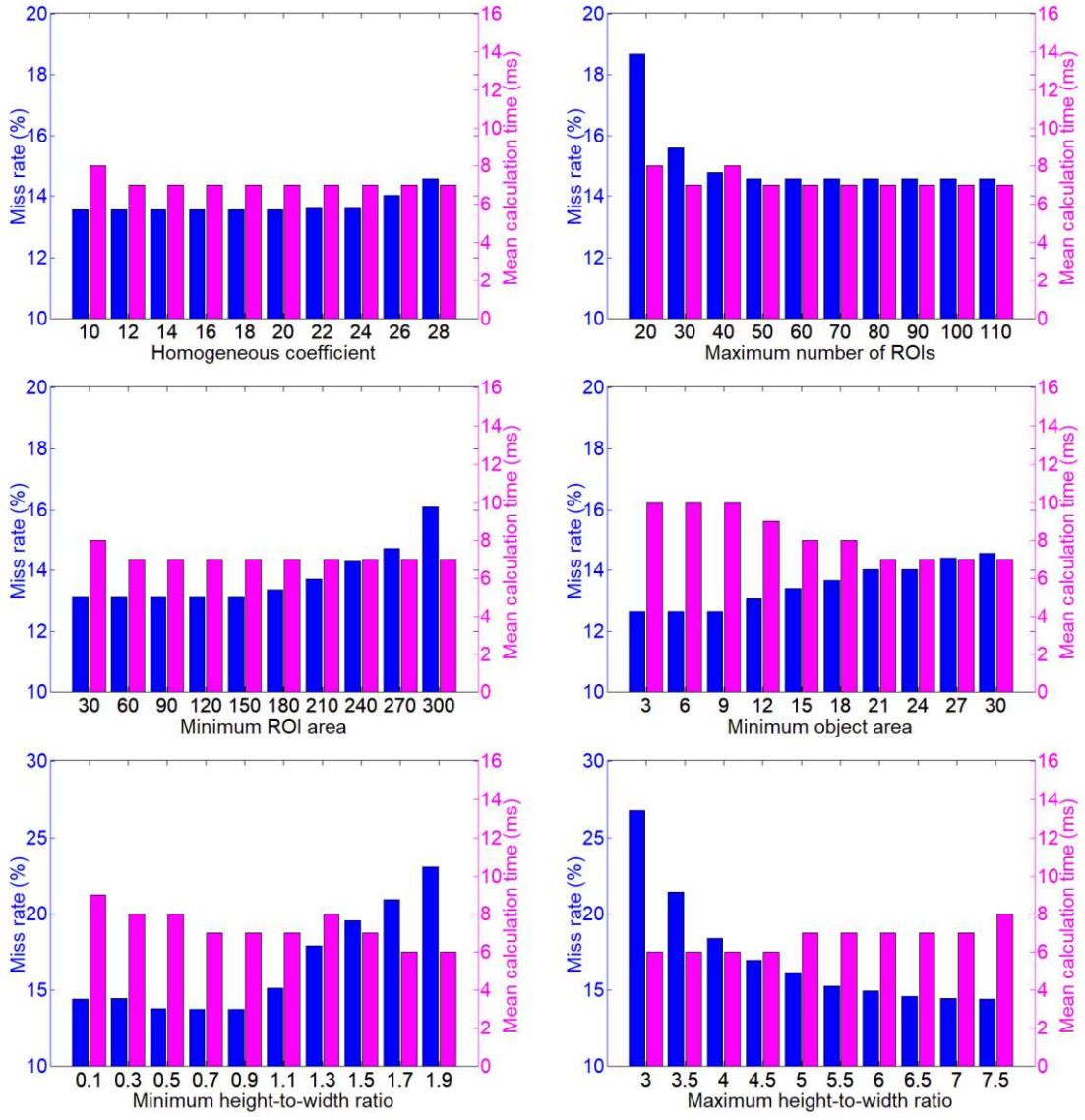


Figure 26. Miss rate [%] and mean calculation time [ms] for different values of candidates selection parameters for CVC-14 dataset, part 1

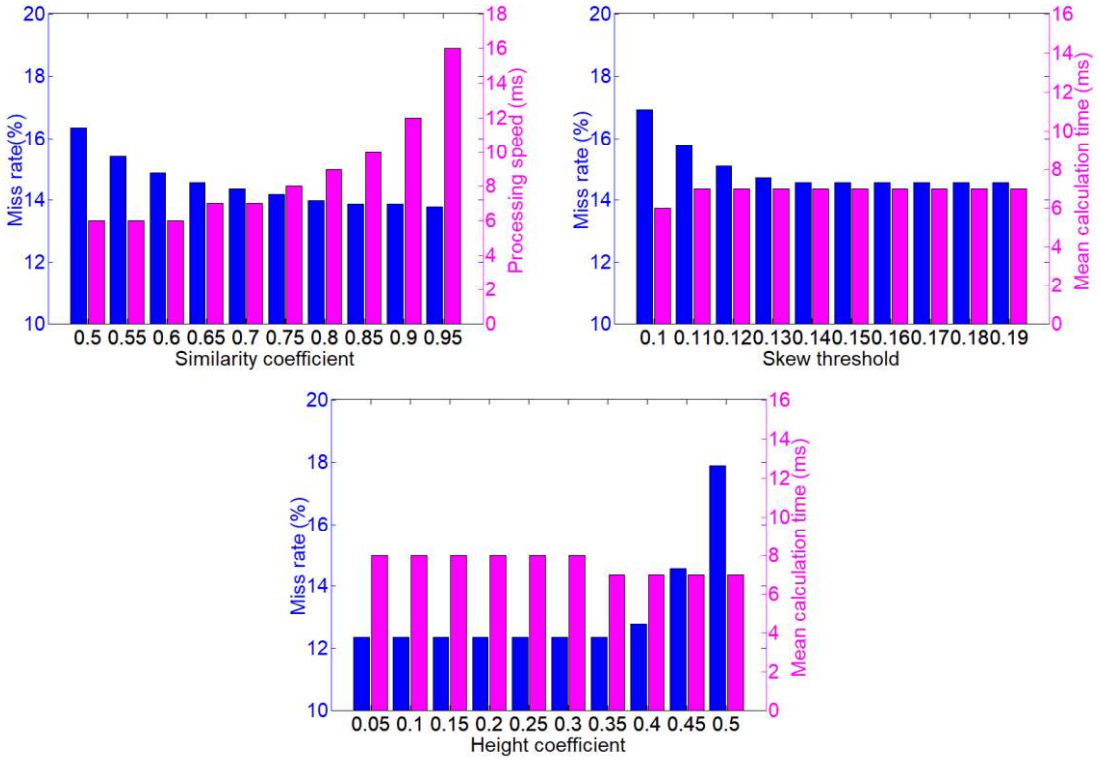


Figure 27. Miss rate [%] and mean calculation time [ms] for different values of candidates selection parameters for CVC-14 dataset, part 2

Table 16. Selected parameters values for the CVC-14 dataset

Name of the parameter	**Symbol**	***Initial values***	***balanced***	***best accuracy***
minimum object area	A_{init}	30 pixels	15 pixels	9 pixels
skew threshold	α_{η}	0.14	0.13	0.14
minimum ROI area	A_{ROI}	250 pixels	180 pixels	150 pixels
minimum height-to-width ratio	$\alpha_{\text{HW}_{\text{min}}}$	0.9	0.9	0.9
maximum height-to-width ratio	$\alpha_{\text{HW}_{\text{max}}}$	6.5	6.5	6.5
similarity coefficient	α_{sim}	0.65	0.65	0.8
homogeneous coefficient	α_{σ}	28	24	20
height coefficient	α_{h}	0.45	0.4	0.35
maximum number of ROIs per one image	$l_{\text{ROIs}_{\text{max}}}$	50	50	50

Table 17. Best results obtained for CVC-14 dataset with proposed ROI generation with adjusted values of candidates selection parameters and with various maximum number of ROIs per one image

		$l_{\mathrm{ROIs_{max}}} = 50$				$l_{\mathrm{ROIs_{max}}} = 150$			
		Double thresholding		Triple thresholding		Double thresholding		Triple thresholding	
		balanced	*best accuracy*	*balanced*	*best accuracy*	*balanced*	*best accuracy*	*balanced*	*best accuracy*
FIXED	T_{L}	155	155	130	130	155	155	130	130
	T_{H}	205	205	205	205	205	205	205	205
	MR [%]	6.7	6.2	5.6	8.7	5.7	3.2	2.2	1.2
	PR	27.9	38.1	34.9	41.2	30.6	57.5	48.5	85.1
	MCT [ms]	11	34	30	111	11	22	24	82
OTSU	α_{caf}	60	60	45	45	60	60	45	45
	β	24	24	40	40	24	24	40	40
	MR [%]	7.3	6.6	6.7	9.9	6.8	4.1	2.9	1.4
	PR	26.9	38.2	35	40.1	28.5	53.4	49.4	85.8
	MCT [ms]	11	27	32	104	10	18	24	73

(*) The mean calculation time MCT was calculated for single-core of Intel Core i7-870 CPU

Results presented in Figure 26 and Figure 27 show to what extent the values parameters of candidates selection affect the MR and MCT coefficients. Typically, lowering the MR value by changing one of the parameters of candidates selection increases the values of MCT and PR parameters. However, only for the $l_{\mathrm{ROIs_{max}}}$ parameter, reducing the maximum possible number of ROIs increases both the MR and the MCT.

In many cases, the proposed parameter values were not perfectly matched, and the MR value can be reduced. Therefore, two sets of parameters were proposed and presented in Table 16 (based on the results presented in Figure 26 and Figure 27). The first set of selected values (*balanced*) was selected to ensure both: high accuracy of pedestrian detection (low MR parameter value) and high computational efficiency (low MCT parameter value). The second set of parameters (*best accuracy*) was selected to achieve the lowest possible MR.

After selecting new sets of values of parameters for the candidate selection, experiments were repeated with the double and triple thresholding for the best threshold values only and are presented in Table 17.

As a result, the value of the MR decreases significantly even to 5.6% (for the triple thresholding with fixed threshold and *balanced* settings). On the other hand, the values

of MCT increased significantly compared to the initial settings of the candidates selection (from 16 ms to 30 ms).

The results are not better only with the triple thresholding with the *best accuracy* settings. After performing additional tests with a higher value of $l_{\mathrm{ROIs}_{\max}}$ equal to 150, the MR decreased also for the *best accuracy* settings reaching even a very low MR value of 1.2% (for fixed triple thresholding with *best accuracy* settings). Moreover MR value decreased significantly for all configurations.

It can also be seen that in all performed experiments conducted for CVC-14 dataset, the segmentation with fixed thresholds still achieves lower MR values than the Otsu-based thresholds, but the difference decreased to about 1%.

When evaluating the final results (for $l_{\mathrm{ROIs}_{\max}}$ = 150, presented in Table 17), the reasonable values of MR, PR and MCT are obtained using the triple thresholding technique with *balanced* settings. The algorithm for these settings is several times faster than for the *best accuracy* settings, MCT decreases from 82 ms to 24 ms, *PR* decreases from 85.1 to 48.5, and the achieved MR value is equal to 2.2% and it is only 1% above the lowest-achieved result (compared to the triple thresholding technique with the *best accuracy*). Few illustrative images with marked ROIs obtained with the proposed ROI generation approach are shown in Figure 28.

Figure 28. Illustrative examples of proposed ROI generation stage on CVC-14 dataset, from the left: input thermal image, image segmented with T_L threshold (with marked ROIs after first thresholding), image segmented with T_H threshold (with marked ROIs after second thresholding), thermal image with marked set of final ROIs after regions enlargement and candidates selection

3.8.3. Calibration on KAIST dataset

Similar, as for the CVC-14 dataset, experiments were performed for the KAIST dataset. Set 09 (campus) was selected as the most representative for the initial experiments (from 3 available night-time test sets, as presented in Section 2.2.6) with the selection of thresholds and parameters values for candidates selection. The results for the remaining KAIST test sequences are also presented (with the finally selected thresholds and sets of parameters of candidates selection process) at the end of the section.

The initial values of candidates selection parameters for experiments were selected experimentally and their values are presented in Table 18.

Table 18. Pre-selected parameters values for the KAIST dataset

Type	Name of the parameter	Symbol	Initial values
Initial ROIs selection	minimum object area	A_{init}	10 pixels
	skew threshold	α_{η}	0.14
Candidates selection	minimum ROI area	A_{ROI}	40 pixels
	minimum height-to-width ratio	$\alpha_{\text{HW}_{\text{min}}}$	0.7
	maximum height-to-width ratio	$\alpha_{\text{HW}_{\text{max}}}$	6.5
	similarity coefficient	α_{sim}	0.65
	homogeneous coefficient	α_{σ}	3
	height coefficient	α_{h}	0.15
	maximum number of ROIs per one image	$l_{\text{ROIs}_{\text{max}}}$	50

Selection of thresholds

A selection of thresholds was performed for the proposed ROI generation technique. The results of the single thresholding experiments are shown in Figure 29 and Figure 30, and the results for the double and triple thresholding are collected in Tables 19-26. The summary of the best results for the double and triple thresholding is presented in Table 27.

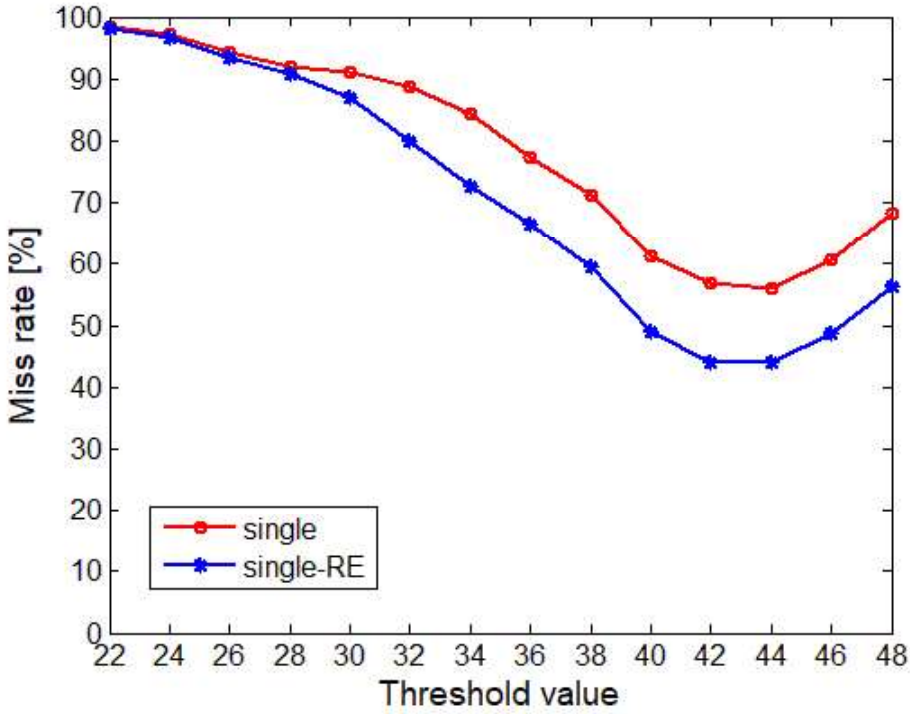


Figure 29. Miss rate for various thresholds for the KAIST dataset with a single fixed threshold (in legend: single – denotes segmentation without regions enlargement, single-RE – denotes segmentation with regions enlargement)

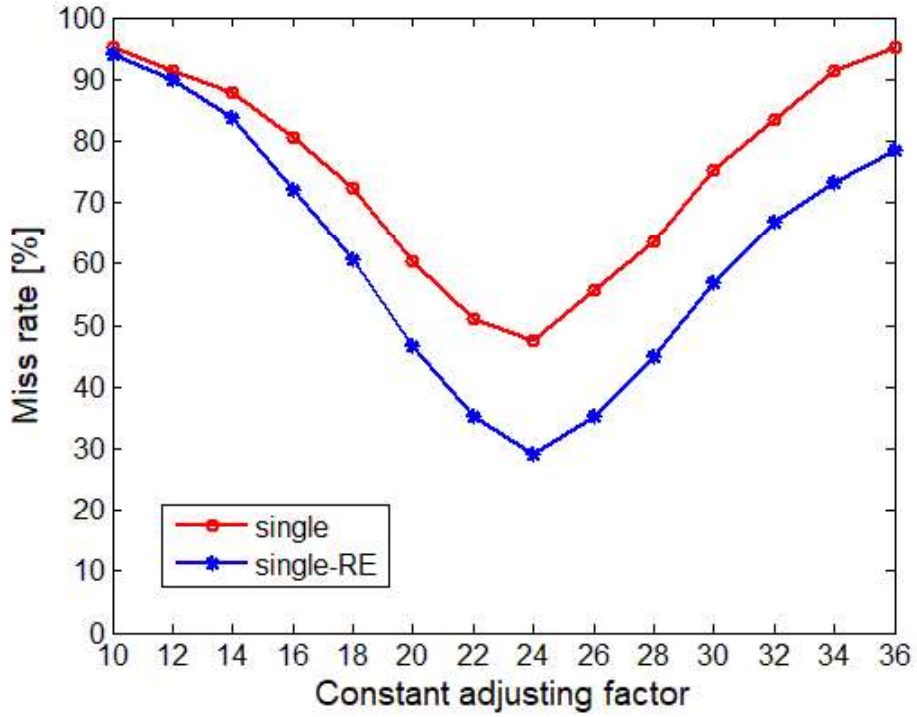


Figure 30. Miss rate for various thresholds for the KAIST dataset with a single Otsu-based threshold (in legend: single – denotes segmentation without regions enlargement, single-RE – denotes segmentation with regions enlargement)

Table 19. Miss rate [%] for various thresholds for the KAIST dataset with double fixed thresholds and with regions enlargement

T_L \ T_H	42	44	46	48	50	52	54	56	58	60	62	64	66
16	43.4	43.6	48.3	56.2	62.2	66.7	70.8	76.1	83.0	88.7	92.1	95.5	97.4
18	43.4	43.6	48.3	56.2	62.3	66.6	70.7	76.0	83.0	88.7	92.1	95.5	97.4
20	43.1	43.2	47.8	55.7	61.8	66.2	70.3	75.7	82.7	88.4	91.8	95.3	97.2
22	41.4	40.8	45.4	53.7	59.9	64.5	68.7	74.0	81.2	86.8	90.2	93.6	95.5
24	40.1	39.6	44.4	52.4	58.7	63.1	67.3	72.7	79.8	85.5	88.9	92.3	94.2
26	38.2	36.8	41.0	48.5	54.3	59.2	63.4	69.0	76.2	82.1	85.6	89.1	90.9
28	35.4	34.3	36.8	42.7	48.3	53.9	58.9	64.9	72.2	78.3	81.8	85.2	87.0
30	33.1	31.3	32.5	38.0	43.4	49.1	54.1	60.0	67.9	73.6	77.3	81.2	83.6
32	28.6	26.5	28.0	33.1	37.9	42.4	46.6	51.7	58.8	65.1	69.1	73.1	75.2
34	30.5	27.5	27.9	30.7	34.0	37.2	41.7	47.3	54.5	59.8	63.7	67.4	69.3
36	33.5	29.2	28.0	28.5	31.1	34.8	38.5	43.0	49.4	54.4	57.5	60.5	62.8
38	35.9	30.5	28.8	28.2	30.6	33.4	36.0	39.4	45.0	49.5	52.4	55.2	57.1
40	39.7	32.8	30.1	27.2	27.8	29.2	30.5	32.7	36.8	40.3	42.7	44.7	46.5

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 20. Miss rate [%] for various thresholds for the KAIST dataset with double fixed thresholds and without regions enlargement

T_L \ T_H	42	44	46	48	50	52	54	56	58	60	62	64	66
16	56.8	56.0	60.7	68.3	76.8	84.5	91.0	96.1	99.4	99.9	99.9	99.9	99.9
18	56.8	55.9	60.6	68.2	76.7	84.4	90.9	96.0	99.3	99.9	99.9	99.9	99.9
20	56.6	55.7	60.5	68.1	76.5	84.3	90.8	95.8	99.1	99.8	99.8	99.8	99.8
22	55.2	54.3	59.1	66.7	75.2	82.9	89.4	94.4	97.8	98.4	98.4	98.4	98.4
24	53.9	53.0	57.8	65.4	73.9	81.6	88.1	93.2	96.5	97.1	97.1	97.1	97.1
26	51.1	50.2	54.9	62.6	71.0	78.8	85.3	90.3	93.6	94.2	94.2	94.2	94.2
28	48.8	47.9	52.6	60.3	68.7	76.5	82.9	88.0	91.3	91.9	91.9	91.9	91.9
30	48.5	47.6	52.2	59.5	67.8	75.6	82.1	87.1	90.4	91.0	91.0	91.0	91.0
32	47.4	46.2	50.2	57.3	65.6	73.3	79.8	84.8	88.2	88.8	88.8	88.8	88.8
34	48.2	46.2	48.9	53.8	61.2	68.8	75.2	80.3	83.6	84.2	84.2	84.2	84.2
36	48.5	44.9	45.8	48.7	55.5	62.4	68.4	73.5	76.8	77.4	77.4	77.4	77.4
38	49.5	44.8	44.2	46.3	52.0	57.7	63.1	67.5	70.7	71.2	71.2	71.2	71.2
40	52.1	45.2	42.7	42.3	45.7	50.1	54.5	58.1	60.7	61.1	61.1	61.1	61.1

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 21. Miss rate [%] for various thresholds for the KAIST dataset with double Otsu-based thresholds and with regions enlargement

α_{caf} \ β	1	2	3	4	5	6	7	8	9	10	11	12	13
10	91.1	88.6	86.2	83.2	77.5	71.5	65.9	60.3	54.9	46.0	40.4	34.9	29.8
12	85.9	82.1	76.6	70.8	65.6	59.6	54.2	45.2	39.7	35.2	29.5	28.6	31.2
14	76.5	70.3	64.7	58.3	52.1	43.9	38.7	33.4	28.7	27.8	30.7	34.6	39.5
16	64.5	57.5	52.9	45.3	37.1	30.3	24.7	24.1	27.6	32.1	38.0	43.8	49.3
18	51.7	41.3	35.7	29.0	23.5	21.9	23.9	26.5	32.8	39.0	45.5	53.7	61.3
20	37.4	28.4	19.0	16.6	19.2	23.6	28.0	33.3	39.7	47.4	55.1	60.7	66.9
22	27.3	20.3	17.8	16.7	18.0	23.9	30.9	39.5	44.2	49.8	58.4	64.4	69.0
24	24.4	20.7	19.8	20.7	23.1	27.9	32.5	39.2	45.9	53.5	58.1	63.9	70.6
26	30.1	25.7	22.8	22.2	23.7	27.1	31.8	36.8	41.1	48.2	55.9	63.7	69.9
28	39.7	34.3	29.7	26.0	24.4	25.1	26.9	31.3	38.6	44.0	50.8	57.3	65.4
30	49.7	44.2	39.3	34.0	29.4	25.4	24.7	28.1	32.3	37.5	46.6	52.5	59.0
32	62.6	56.0	48.8	43.2	38.5	33.5	29.6	26.8	27.2	31.8	37.2	42.9	52.2
34	69.4	65.2	61.5	55.9	49.4	44.2	39.5	34.4	30.6	28.6	29.3	34.4	40.0
36	75.0	72.4	69.6	66.2	62.6	56.6	50.2	44.8	40.1	35.1	31.4	28.9	29.8

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 22. Miss rate [%] for various thresholds for the KAIST dataset with double Otsu-based thresholds and without regions enlargement

α_{caf} \ β	1	2	3	4	5	6	7	8	9	10	11	12	13
10	93.3	90.6	88.6	86.5	84.1	80.2	76.6	72.2	68.3	60.5	55.2	51.0	47.2
12	88.2	85.4	82.1	78.0	74.8	70.5	67.5	60.2	55.0	51.0	47.2	47.6	51.8
14	82.3	76.5	72.3	68.6	65.1	57.6	52.8	49.0	46.5	47.4	51.7	55.7	58.9
16	74.2	67.3	62.0	53.9	49.3	46.3	43.3	44.2	49.4	53.7	58.2	63.4	69.0
18	63.5	52.5	46.6	42.2	38.9	39.7	45.5	50.8	54.9	60.3	66.7	73.3	79.2
20	50.6	41.9	34.8	34.8	39.9	45.5	49.5	55.5	62.8	70.4	75.9	80.2	85.9
22	42.5	36.3	37.6	39.0	41.9	47.4	54.9	63.6	69.7	74.9	82.0	86.5	89.5
24	42.1	38.4	37.5	39.8	46.5	51.9	58.1	64.7	73.2	79.4	83.4	86.5	89.9
26	50.5	44.4	41.1	41.6	44.1	48.8	58.0	63.9	70.3	75.7	81.1	85.5	88.1
28	58.7	55.0	50.3	45.5	44.0	46.3	50.5	56.4	64.6	69.8	75.1	79.8	84.4
30	68.9	63.4	58.7	55.2	50.9	46.9	46.2	49.3	53.7	59.8	67.9	72.0	76.6
32	79.8	75.2	69.0	63.6	58.9	55.5	51.7	47.6	47.1	50.8	55.0	60.5	68.3
34	87.9	83.5	79.8	75.3	69.2	63.7	58.9	55.6	51.7	47.6	47.2	51.0	55.2
36	93.4	91.2	88.3	83.6	79.9	75.3	69.2	63.7	58.9	55.7	51.8	47.6	47.2

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 23. Miss rate [%] for various thresholds for the KAIST dataset with triple fixed thresholds and with regions enlargement

T_L \ T_H	42	44	46	48	50	52	54	56	58	60	62	64	66
16	33.8	30.9	28.7	33.0	36.2	37.1	39.5	42.6	46.2	49.1	45.6	44.2	43.5
18	32.8	27.2	27.7	32.3	33.6	35.5	38.0	40.2	44.6	43.4	42.1	42.5	41.5
20	29.4	26.1	28.0	29.8	30.7	33.5	35.7	38.0	38.3	39.0	40.3	40.0	40.8
22	28.1	26.7	26.7	27.2	29.5	31.7	33.8	32.4	34.5	36.8	36.5	37.6	39.0
24	28.0	25.9	25.7	26.2	28.2	30.5	29.1	29.5	32.3	33.8	34.8	36.7	40.6
26	29.8	26.1	26.5	26.0	27.2	27.0	25.9	27.1	29.3	31.4	32.5	36.8	39.4
28	29.4	26.0	24.4	23.9	23.2	23.1	23.3	23.5	27.1	28.9	32.1	34.8	40.0
30	29.5	24.7	22.6	19.6	19.4	19.4	20.0	22.2	25.6	28.6	30.1	34.7	37.8
32	25.8	21.2	18.4	15.9	15.3	15.3	16.8	18.8	22.6	24.5	30.0	32.6	35.9
34	29.0	23.9	21.8	18.6	17.5	17.9	19.2	21.1	24.3	27.3	29.6	31.8	34.0
36	32.2	27.4	24.5	21.8	20.4	21.1	22.5	23.3	25.7	27.0	29.1	31.1	33.0
38	36.0	29.3	26.8	23.8	22.5	23.1	24.2	25.0	26.3	28.4	30.7	32.2	33.5
40	40.2	33.5	29.4	26.2	25.2	25.7	26.4	25.4	26.1	27.4	28.3	29.4	29.7

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 24. Miss rate [%] for various thresholds for the KAIST dataset with triple fixed thresholds and without regions enlargement

T_L \ T_H	42	44	46	48	50	52	54	56	58	60	62	64	66
16	49.2	47.6	49.7	57.3	64.1	68.8	70.9	73.5	73.5	71.2	64.6	61.1	57.7
18	48.5	45.2	50.1	56.4	61.1	64.5	68.4	70.1	70.6	64.5	61.1	57.6	56.8
20	46.2	45.9	50.0	53.5	57.1	62.2	65.0	67.3	63.9	60.9	57.4	56.6	55.7
22	45.9	45.2	47.3	48.6	53.9	58.0	61.5	59.7	59.1	56.0	55.2	54.3	54.3
24	46.1	43.9	44.1	46.2	50.5	54.8	54.3	55.1	54.5	53.9	53.0	53.0	54.9
26	43.7	40.3	41.1	41.9	46.4	46.7	48.8	49.7	50.8	50.2	50.2	52.1	54.9
28	41.5	38.2	37.1	38.6	39.9	42.2	44.4	46.8	47.6	47.9	49.8	52.6	57.1
30	42.3	37.3	36.6	35.8	37.8	40.3	43.9	46.1	47.5	49.4	52.2	56.4	59.5
32	41.4	36.9	34.4	33.6	36.1	39.3	42.7	44.9	47.7	50.2	54.4	57.3	61.4
34	44.4	38.3	36.0	35.0	37.4	40.1	43.2	46.4	48.8	51.7	53.8	57.2	61.2
36	45.7	39.8	36.5	36.3	37.2	39.3	42.6	45.2	47.2	48.7	51.8	55.5	58.7
38	48.8	41.6	38.9	37.2	37.3	39.7	42.3	45.2	46.3	48.8	52.0	54.7	57.7
40	52.4	45.0	40.7	37.6	37.5	38.8	41.7	42.1	43.6	45.8	47.7	50.1	52.6

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 25. Miss rate [%] for various thresholds for the KAIST dataset with triple Otsu-based thresholds and with regions enlargement

α_{caf} \ β	1	2	3	4	5	6	7	8	9	10	11	12	13
10	90.6	87.9	84.8	81.8	76.1	70.0	64.4	58.2	52.5	44.2	37.6	30.1	23.1
12	85.5	80.9	75.6	70.0	65.1	58.5	52.9	44.9	37.5	30.8	24.5	21.4	23.0
14	75.4	68.6	62.6	56.2	50.5	42.9	36.3	29.8	22.9	20.2	20.9	23.1	26.8
16	63.5	56.6	50.6	41.9	33.3	26.2	19.4	16.5	17.0	18.3	20.6	23.9	27.1
18	51.7	41.1	33.5	26.3	20.4	16.6	14.8	14.7	16.9	20.4	23.3	27.3	29.9
20	38.3	29.5	20.4	17.3	17.5	18.3	18.7	19.9	19.9	22.0	22.9	25.2	27.1
22	28.0	21.8	17.7	15.5	13.7	15.0	17.4	18.8	19.7	20.4	20.6	21.3	21.4
24	24.4	20.7	18.6	17.4	15.7	14.5	13.4	14.5	16.5	17.9	17.9	19.2	18.9
26	29.9	25.5	21.9	19.7	18.6	18.5	17.1	16.1	14.8	16.7	19.5	22.4	23.3
28	39.7	34.2	29.6	25.3	21.9	20.4	19.7	19.7	19.9	20.5	20.4	23.6	26.7
30	49.7	44.1	39.1	33.6	28.5	24.1	21.9	21.4	21.7	23.7	26.1	27.8	29.1
32	62.5	55.8	48.7	42.8	38.0	32.9	28.7	25.1	23.7	23.8	24.9	27.0	30.1
34	69.3	65.0	61.3	55.3	48.9	43.6	38.7	33.6	29.4	26.1	24.5	25.1	26.4
36	74.8	72.0	68.8	65.1	61.3	55.3	48.6	43.2	38.3	33.1	29.0	25.4	24.2

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 26. Miss rate [%] for various thresholds for the KAIST dataset with triple Otsu-based thresholds and without regions enlargement

α_{caf} \ β	1	2	3	4	5	6	7	8	9	10	11	12	13
10	93.0	90.1	87.8	85.1	81.4	76.9	73.1	68.6	64.4	56.3	50.6	46.3	42.4
12	88.2	85.1	81.0	75.8	71.6	66.6	62.2	53.7	48.0	43.7	39.7	39.7	43.7
14	82.2	76.0	70.5	64.7	60.0	51.0	45.3	40.7	37.5	37.7	41.6	45.5	48.4
16	74.0	66.4	58.9	48.5	41.9	36.9	32.6	32.3	36.0	39.3	42.9	47.1	51.8
18	63.0	51.1	44.0	36.9	30.6	29.2	31.9	35.3	37.8	41.5	45.8	50.1	54.1
20	50.4	40.8	31.9	28.5	29.4	29.9	31.0	33.3	36.4	40.6	43.4	45.9	49.2
22	42.9	35.0	31.5	29.2	27.5	28.6	30.6	32.9	34.8	36.3	39.2	41.6	43.6
24	42.1	37.4	34.2	32.7	32.5	31.8	31.4	32.6	34.5	36.5	37.5	38.9	40.7
26	50.5	44.4	39.9	37.5	36.1	36.5	38.7	38.5	39.0	40.7	43.0	45.3	46.4
28	58.6	55.0	50.2	44.3	40.4	39.1	38.9	39.8	43.4	44.1	45.3	47.4	49.9
30	68.9	63.3	58.5	54.9	50.1	45.0	42.0	41.6	42.5	44.8	50.0	51.9	54.4
32	79.8	75.1	68.8	63.4	58.5	55.0	50.3	45.5	43.2	43.6	45.5	48.8	54.6
34	87.9	83.3	79.7	75.2	69.0	63.6	58.8	55.2	50.7	46.1	44.7	46.3	49.1
36	93.4	91.2	88.0	83.5	79.8	75.3	69.1	63.6	58.8	55.3	51.2	46.9	45.9

Scale
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0
90.0
100.0

Table 27. Best experimental results (based on lowest MR value) obtained for the KAIST dataset with the proposed ROI generation with double and triple thresholding and with region enlargement

		Double thresholding				Triple thresholding			
FIXED	T_L	32	40	34	40	32	32	32	32
	T_H	44	48	44	50	50	52	48	54
	MR [%]	26.5	27.2	27.5	27.8	15.3	15.3	15.9	16.8
	PR	14.7	13.3	15.4	12.5	19.1	18.2	19.9	17.6
	MCT [ms]	2	1	2	1	3	2	3	2
OTSU	α_{caf}	20	22	22	22	24	22	24	24
	β	4	4	3	5	7	5	6	8
	MR [%]	16.6	16.7	17.8	18	13.4	13.7	14.5	14.5
	PR	17.8	16.5	17.1	15.6	19.9	20.9	19.9	19.8
	MCT [ms]	3	2	2	2	3	4	3	3

(*) The mean calculation time MCT was calculated for single-core of Intel Core i7-870 CPU

As in the case of the CVC-14 dataset, the presented results of the selection of thresholds for the KAIST dataset (Figure 29 and Figure 30, Tables 19-26) show that the accuracy of the proposed ROI generation procedure with the double and triple thresholding is much higher than with the single thresholding. The difference in the achieved MR values equals to 16.6% (between lowest achieved MR values for single and triple thresholding). Therefore, no further analysis for this dataset will be performed for the single thresholding.

It could be noticed also that the regions enlargement technique has the greatest impact on the MR coefficient (similarly as in the case of CVC-14). With this technique it was possible to lower the MR parameter, e.g. for double thresholding with fixed thresholds, the MR value decreased from 46.2% to 26.5% for thresholds values T_L = 32 and T_H = 44 (see Table 19 and Table 20), for triple thresholding with fixed thresholds, the MR value decreased from 36.1% to 15.3% for thresholds values T_L = 32 and T_H = 50 (see Table 23 and Table 24). A similar tendency can be observed in the case of Otsu-based thresholds.

The use of the triple thresholding technique also reduces MR to some extent compared to the double thresholding. The difference is equal to 11.2% (change from 26.5% to 15.3%) for fixed thresholds and equal to 3.2% (change from 16.6% to 13.4%) for Otsu-based thresholds. In addition, it is also important that the MR value is less dependent on the adjustment of the threshold values for the triple thresholding (see Tables 19-26). Moreover, the mean processing time MCT for the triple thresholding increases compared to the double thresholding, from approximately 2 ms to 4 ms, and the number of selected ROIs per frame also increases from approximately 15 to 20.

Contrary to the results obtained for the CVC-14 dataset, the experiments conducted for KAIST show that the use of Otsu-based thresholds approach is more advantageous than the fixed-based thresholds approach, e.g., the MR value for Otsu-based thresholds is lower for triple thresholds and equals to 13.4% compared to 15.3% for fixed values of thresholds (see Table 27).

Candidates selection

In the next step, experiments were conducted to verify the impact of changes in the values of parameters of candidates selection process on the MR and MCT metrics and to select sets of optimal parameters values for double and triple thresholding.

The initial values of tested parameters were selected experimentally and their values are presented in Table 18. Results of the experiments are presented in Figure 31 and Figure 32.

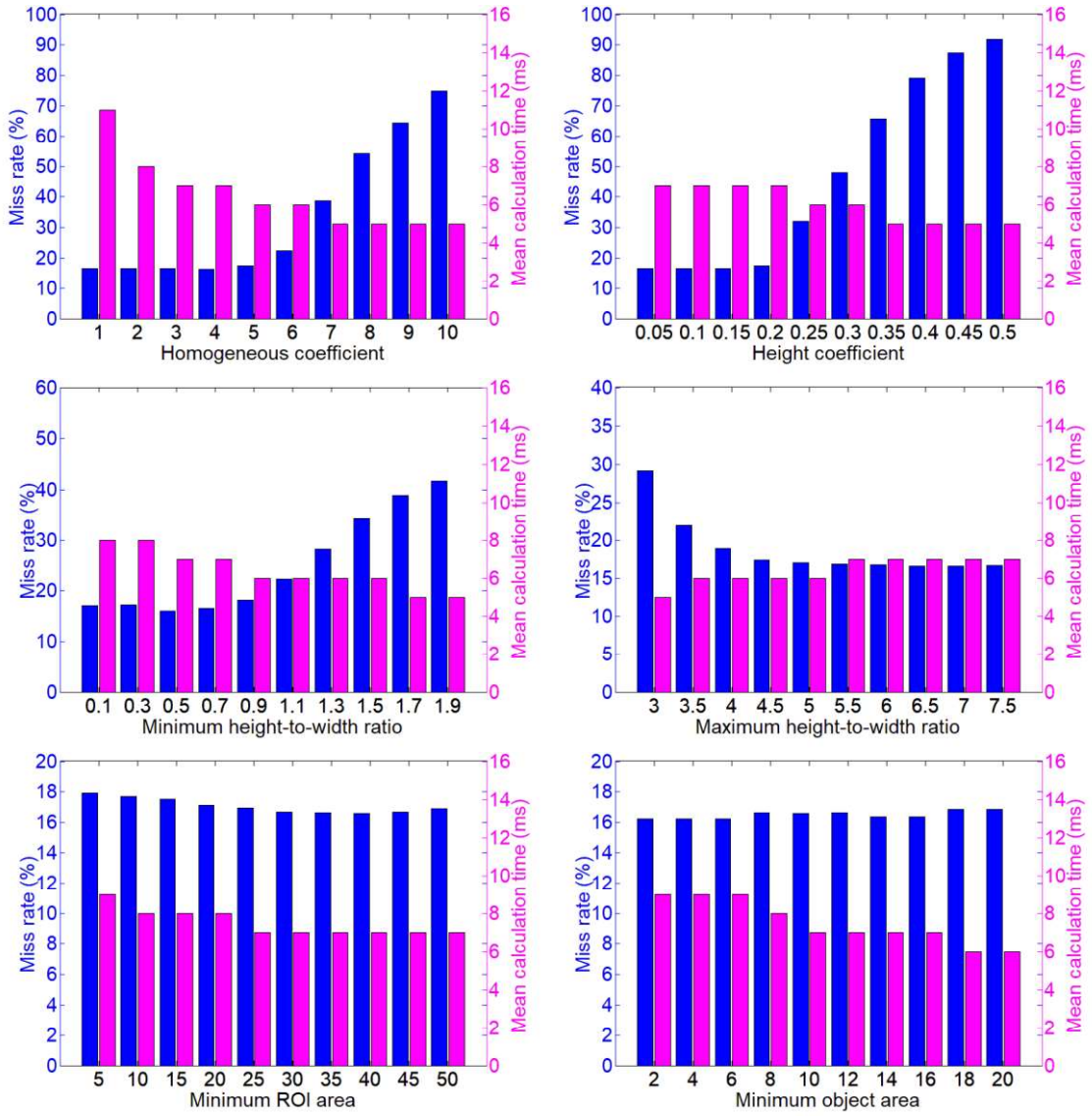


Figure 31. Miss rate [%] and mean calculation time [ms] for different values of candidates selection parameters for KAIST dataset, part 1

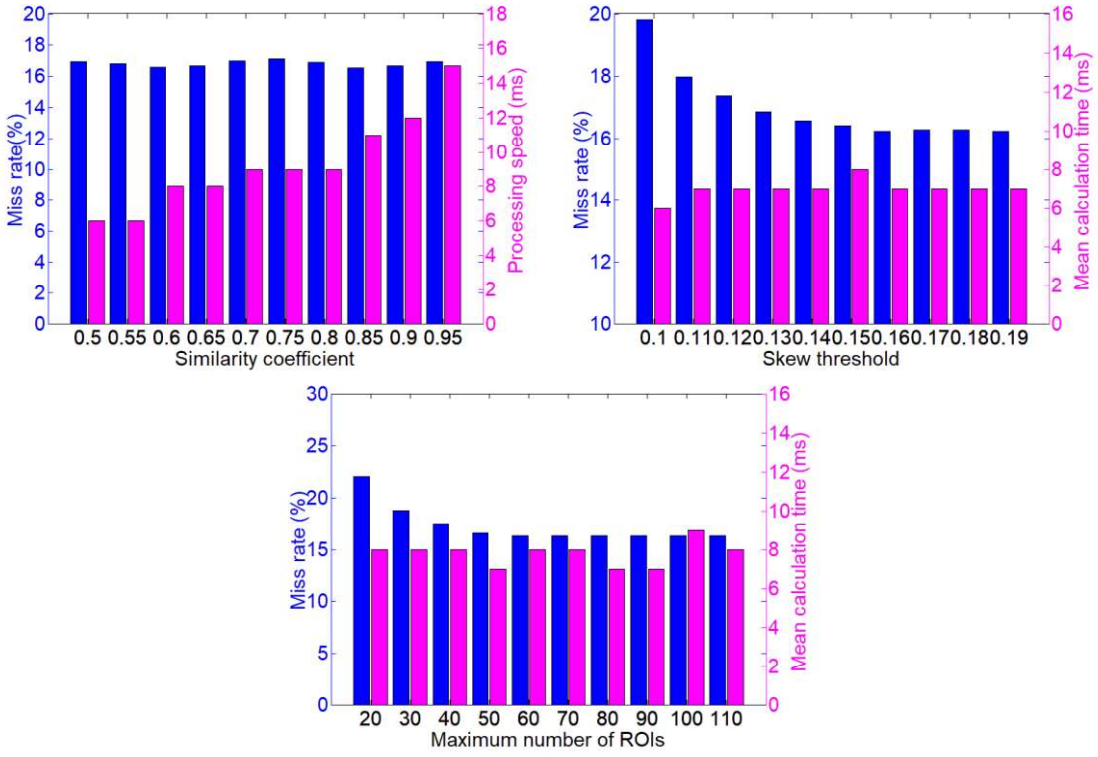


Figure 32. Miss rate [%] and mean calculation time [ms] for different values of candidates selection parameters for KAIST dataset, part 2

Table 28. Selected parameters values for the KAIST dataset

Name of the parameter	**Symbol**	***Initial values***	***balanced***	***best accuracy***
minimum object area	A_{init}	10 pixels	16 pixels	6 pixels
skew threshold	α_{η}	0.14	0.16	0.16
minimum ROI area	A_{ROI}	40 pixels	50 pixels	40 pixels
minimum height-to-width ratio	$\alpha_{\text{HW}_{\min}}$	0.7	0.7	0.5
maximum height-to-width ratio	$\alpha_{\text{HW}_{\max}}$	6.5	6.5	6.5
similarity coefficient	α_{sim}	0.65	0.6	0.6
homogeneous coefficient	α_{σ}	3	4	4
height coefficient	α_{h}	0.15	0.15	0.15
maximum number of ROIs per one image	$l_{\text{ROIs}_{\max}}$	50	50	80

Table 29. Final experimental results obtained for the KAIST dataset (set 09) with proposed ROI generation with adjusted values of candidates selection parameters and with various maximum number of ROIs per one image

		Double thresholding		**Triple thresholding**	
		balanced	*best accuracy*	*balanced*	*best accuracy*
FIXED	T_{L}	32	32	32	32
	T_{H}	44	44	50	50
	MR [%]	26.4	25.5	15.1	11.9
	PR	12.1	15.1	14.7	22.1
	MCT [ms]	2	3	3	4
OTSU	α_{caf}	20	20	24	24
	β	4	4	7	7
	MR [%]	16.6	13.2	13.1	11.1
	PR	11.9	18.3	14.5	22
	MCT [ms]	2	3	3	3

(*) The mean calculation time MCT was calculated for single-core of Intel Core i7-870 CPU

The results presented in Figure 31Figure 32 and Figure 32 show similar trends as in the case of the CVC-14 dataset. Typically, lowering the MR value by changing one of the parameters of candidates selection increases the values of MCT and PR parameters. However, only for the $l_{\mathrm{ROIs}_{\max}}$ parameter, reducing the maximum possible number of ROIs increases the MR value.

In few cases, the proposed parameter values were not perfectly matched, and the MR value can further be reduced. Therefore, two sets of parameters were proposed (as before for the CVC-14 dataset, based on the results presented in Figure 32). The first set of selected values (*balanced*) was selected to ensure both: high accuracy of pedestrian detection (low MR parameter value) and high computational efficiency (low MCT parameter value). The second set of parameters (*best accuracy*) was selected to achieve the lowest possible MR.

After selecting new sets of candidate selection parameters (included in Table 28), experiments were repeated with the double and triple thresholding for the best threshold values only and are presented in Table 29.

As a result, the value of the MR decreases to 11.1% (for the Otsu-based triple thresholding with the *best accuracy* settings). The values of MCT do not increase for double thresholding or even decreased for triple thresholding compared to the results obtained for the initial settings of the candidates selection.

The results does not improve significantly only for the double thresholding with fixed thresholds. In addition, triple thresholding generally gives better results than

double thresholding, e.g., the MR difference is equal to 2.1% for Otsu-based thresholds (11.1% for the triple thresholding and 13.2% for the double thresholding) while achieving slightly worse MCT, and PR values.

It can also be seen that in all performed experiments, the segmentation with Otsu-based thresholds still achieves lower MR values than with the fixed thresholds, but the difference decreased to about 1%.

In the next step, experiments were performed for all KAIST test sets (set 09, set 10, and set 11 described in Chapter 3.6). These experiments were conducted for the triple thresholding with Otsu-based thresholds. The results are presented in Table 30.

The obtained results for subset 09 (campus) and subset 10 (roadway) are similar (MR = 11.1% and 11.0% for the *best accuracy* settings). Higher MR values were obtained for the city center, where the thermal contrast is lower due to the urban surroundings. Therefore, additional tests were performed with a higher parameter value $l_{\mathrm{ROIs_{max}}}$ equal to 150. As a result, MR significantly decreased for set 10 (to 2.8%) and set 11 (from 31.1% to 12.7%) for the *best accuracy* settings.

When evaluating the final results (for $l_{\mathrm{ROIs_{max}}}$ = 150, presented in Table 30), the most balanced values of MR, PR and MCT are obtained using the triple thresholding technique with *balanced* settings. The algorithm for these settings is almost two times faster, MCT decreases from 20 ms to 11 ms (for average results), PR decreases from 66.1 to 53.5, and the achieved MR parameter is equal to 10.1%, and is by 1.3% worse than the best result (compared to the triple thresholding with the *best accuracy* settings). Few illustrative images with marked ROIs obtained with the proposed ROI generation are shown in Figure 33.

Table 30. Final experimental results obtained for all tests subsets: 9, 10, 11 from KAIST dataset with the proposed ROI generation procedure for various maximum number of ROIs per one image

		$l_{\mathrm{ROIs_{max}}} = 80$		$l_{\mathrm{ROIs_{max}}} = 150$	
Set no.	Parameter	*balanced*	*best accuracy*	*balanced*	*best accuracy*
Set 09 (campus)	MR [%]	13.1	11.1	12.8	11.1
	PR	14.5	22.0	14.7	22.0
	MCT [ms]	2	3	3	4
Set 10 (roadway)	MR [%]	14.5	11.0	4.1	2.8
	PR	30.0	52.8	46.4	72.4
	MCT [ms]	7	11	7	10
Set 11 (downtown)	MR [%]	37.1	31.1	13.5	12.7
	PR	32.6	56.2	99.5	103.9
	MCT [ms]	31	45	23	46
Average	MR [%]	21.5	17.7	10.1	8.8
	PR	25.7	43.6	53.5	66.1
	MCT [ms]	14	20	11	20

(*) The mean calculation time MCT was calculated for single-core of Intel Core i7-870 CPU

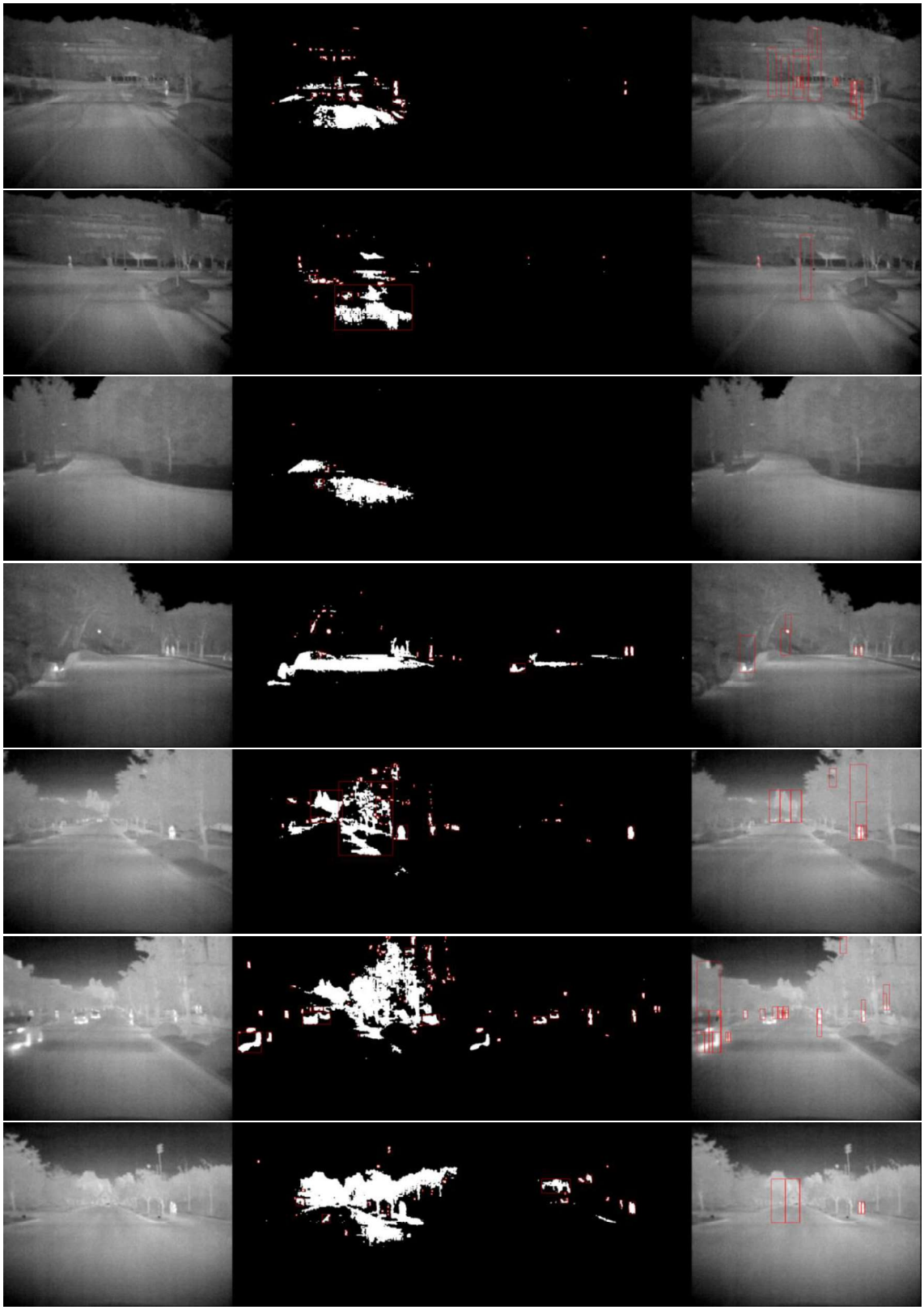


Figure 33. Illustrative examples of proposed ROI generation stage, from left: input thermal image, image segmented with T_L threshold (with marked ROIs after first thresholding), image segmented with T_H threshold (with marked ROIs after second thresholding), thermal image with marked set of final ROIs after regions enlargement and candidates selection

3.9. Summary

The results obtained for both datasets, namely CVC-14 and KAIST allow to conclude that it is possible to accurately and efficiently perform the ROI generation of thermal images at night through the thresholding process. The best results for the proposed ROI generation technique (based on MR value) are presented in Table 31 (for KAIST dataset average results are presented, same as in Table 30). Very low MR values were achieved: 1.2% for the CVC-14 dataset and 8.8% for the KAIST dataset with high computational efficiency (varying from 44 to 347 FPS depending on the settings) obtained using only the CPU.

Table 31. The best-obtained results for CVC-14 and KAIST datasets with the proposed ROI generation procedure

	CVC − 14		**KAIST (average)**	
Parameter	*balanced*	*best accuracy*	*balanced*	*best accuracy*
MR [%]	2.2	1.2	10.1	8.8
PR	48.5	85.1	53.5	66.1
MCT [ms]	24	82	11	20
FPS*	42	12	94	50
FPS**	155	44	347	185

(*) The FPS was calculated for single-core of Intel Core i7-870 CPU
(**) The FPS was calculated for four-core of Intel Core i7-870 CPU

The proposed ROI generation technique produces a very low number of samples per frame compared to other techniques such as the sliding window. On the other hand, it is possible to further increase the accuracy (decrease the MR parameter) of the algorithm, limiting the candidates selection even more (see Table 32). However, even more than thousands of samples per image are created in this case, which significantly slow down the processing of the entire pedestrian detection algorithm. With such a large number of samples for classification, the number of false detections increases at the object classification stage. As a result, the classification threshold will have to be heightened (to obtain a fixed FPPI value), and in consequence, the MR value will increase again. This issue is also considered in Chapter 6.

The proposed ROI generation technique is the most accurate in night conditions. However, when thermal contrast is lower, the accuracy begins to decline, e.g., for the KAIST set 11 (downtown).

The triple thresholding achieves slightly lower MR values than the double thresholding at the cost of slightly less efficiency. The proposed sets of candidates selection (*balanced* and *best accuracy*) also significantly affect the MR, PR, and MCT parameters. The most reasonable results are achieved by the triple thresholding with *balanced* setting, offering a low MR value (slightly higher compared to the *best accuracy* setting, Table 31) and very high computational efficiency (up to 347 FPS

using only CPU). However, the choice of the final settings depends on the type of application.

Image segmentation with Otsu-based thresholds gives better results than fixed-based thresholds for the KAIST dataset but slightly worse for the CVC-14 base. However, the use of Otsu-based thresholds is a more reliable solution due to the possibility of adapting to changes in image dynamics.

Table 32. The MR, PR and MCT values obtained with and without candidates selection for CVC-14 and KAIST datasets with *balanced* settings

	CVC − 14			**KAIST (set 09)**		
	no candidates selection	only initial selection	with candidates selection	no candidates selection	only initial selection	with candidates selection
MR [%]	0.4	0.7	1.2	6.8	10.1	11,1
PR	1608.4	668.2	85.1	604.9	359.2	22.0
MCT [ms]	42	16	82	2	3	4

(*) The mean calculation time MCT was calculated for single-core of Intel Core i7-870 CPU

The experiments also show that some of the candidates selection parameters (similarity coefficient, skew threshold, minimum and maximum height-to-width ratios) are independent from the datasets and camera sensor. Other parameters should be calibrated to the vision system used.

The proposed parameter of a maximum number of ROIs per one image $l_{\mathrm{ROIs}_{max}}$ also significantly affects the operation of the proposed ROI generation algorithm. From the plots of the value of the parameter $l_{\mathrm{ROIs}_{\mathrm{max}}}$ presented in Figure 27 and Figure 32, it follow that increasing this parameter above the value of 50 would not result in a significant decrease in the MR value. However, changes of other parameters values (especially for the *best accuracy*) resulted in a significant increase in the average number of samples per image (PR value). It resulted in more frequent using of proposed additional constraints for individual image frame (as presented in Section 3.7).

Therefore, the technique of limiting of the number of ROIs per frame (if needed) should be used with the high $l_{\mathrm{ROIs}_{\mathrm{max}}}$ value to prevent the entire pedestrian detection algorithm from slowing down too much (by limiting the PR value and consequently the classification time).

4. Adjustment of segmented ROI in thermal night-vision

A quality of the prepared ROIs is very important and significantly affects the effectiveness of classification. All the advanced segmentation techniques, besides the simplest one, i.e., the sliding window technique, match the ROIs to the outer pedestrian edges in the image. The image is divided into areas regarding the edges of objects. This means that pedestrians are very closely matched to the ROIs. However, many problems could arise during segmentation of thermal images, i.e., the uneven level of the observed temperature of one pedestrian and the temporary loss of thermal contrast between the pedestrian and the surroundings.

Inaccurate matching the edges of ROI to the outer edges of the pedestrian may lead to cases of not a whole pedestrian covered with the ROI. Such too small ROIs may be rejected by the classifier. This will finally increase the number of falsely negative results.

Figure 34. Illustrative examples of inaccurate segmented ROIs

Although the proposed ROI generation method (presented in previous Chapter 3) could be very fast, accurate, and producing a low number of false candidates, it was noticed that it sometimes produces ROIs that not including the whole pedestrian. Examples of such ROIs are presented in Figure 34. In these cases, not all body parts of the pedestrians are included in the ROI, whilst primarily the contour edges allow the classifier to determine the shape of a pedestrian. This is especially important in thermal imaging, where the images have few details, and the textures are very poor.

To solve this problem, it was proposed to adjust the segmented ROIs with a scale factor k before the object classification stage. This is done by taking larger area from the image, not just by resizing previously segmented ROI (as presented in Figure 35).

Assuming an original i-th pedestrian candidate obtained with the ROI generation process and described by (x_i, y_i, w_i, h_i), where x_i, y_i are the coordinates of the top left corner of i-th ROI and w_i, h_i are its width and height respectively, the new coordinates of the rescaled ROI (by k scale factor) are calculated as follows:

$$x_{i_{\text{new}}} = x_i + (1-k)/2 \cdot w_i \tag{37}$$

$$y_{i_{\text{new}}} = y_i + (1-k)/2 \cdot h_i \tag{38}$$

$$w_{i_{\text{new}}} = k \cdot w_i \tag{39}$$

$$h_{i_{\text{new}}} = k \cdot h_i \tag{40}$$

As a result, instead of the pedestrian candidate with (x_i, y_i, w_i, h_i) coordinates, the area of pedestrian candidate with the new coordinates $(x_{i_{\text{new}}}, y_{i_{\text{new}}}, w_{i_{\text{new}}}, h_{i_{\text{new}}})$ is taken from the input image. The proposed idea is depicted in Figure 35.

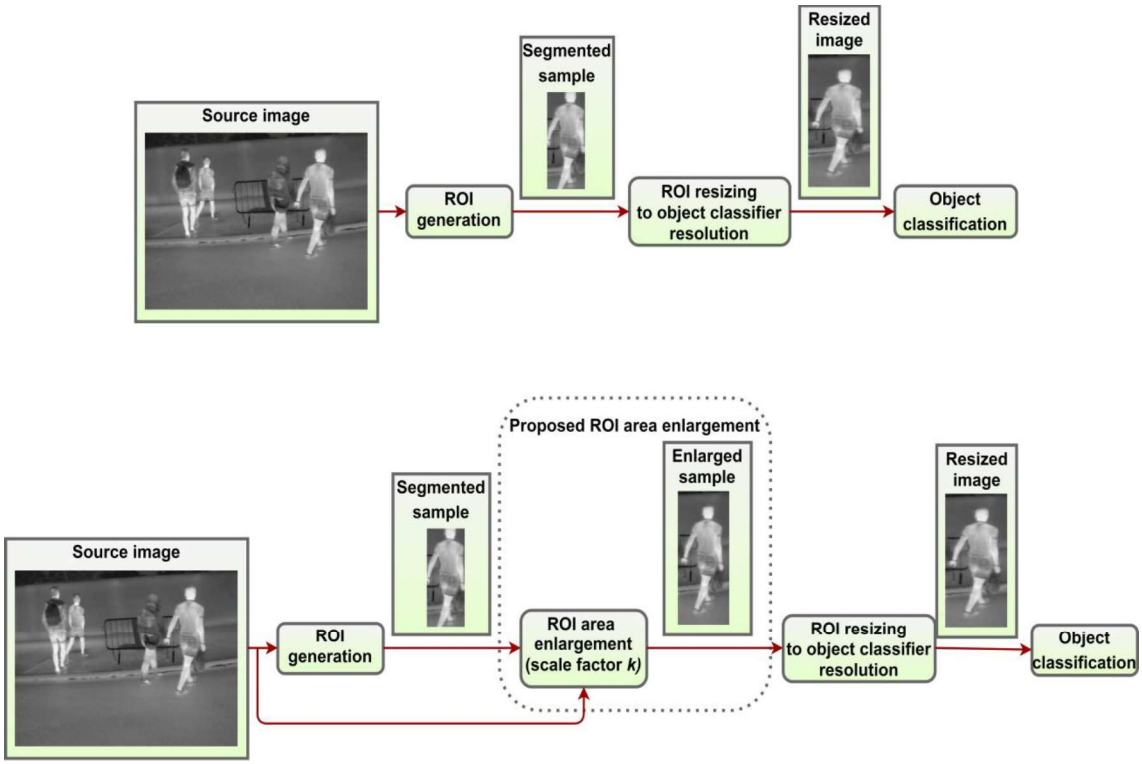


Figure 35. Pedestrian detection procedure diagrams with single-resolution classifier: typical procedure (upper diagram) and modified procedure with additional step of ROI adjustment (lower diagram)

It is important to emphasize that the proposed ROI enlargement will not significantly affect the computational performance of the detection process because before the object classification stage, all ROIs are then resized to the same classifier input resolution. In addition, the number of ROIs that will be classified does not change.

The impact of the proposed method on the performance of the detection process has been tested. The proposed method increases the accuracy of the overall pedestrian detection procedure with little impact on computational performance. The detailed results are presented in Chapter 6, along with the experiments on the proposed pedestrian detection procedure.

5. Tuning of the object classification process

This chapter presents an experimental evaluation of object classification algorithms and their tuning using the proposed universal performance index.

As mentioned in Chapter 2.7, the baseline approach to object classification is to use a classifier with a fixed input resolution. Such fixed classifiers are often used without an adaptation to the resolution of the specific dataset or camera. This is unfortunately a common practice especially in cases, where the structure of a classifier is complicated, e.g., with a deep convolutional neural network (CNN).

To obtain the best results at the object classification stage, it is proposed to look for a compromise input resolution with a proposed universal performance index. The speed of detection and the classification accuracy is taken into account. Using this index, it is possible to select the best input resolution for a particular classifier. The various classifiers were tested with and the results are presented in this chapter. In the experiments, three various baseline detectors were used, namely: histogram of oriented gradients (HOG) with the support vector machine (SVM) classifier, the aggregated channel feature (ACF) detector, and the deep convolutional neural network (CNN).

5.1. Performance Index

The classification stage is one of the crucial parts of the pedestrian detection procedure. Especially in real-time applications with embedded systems (e.g., in cars) this stage must be fast and accurate.

In the literature concerning machine learning, it is possible to find many parameters describing the classifier effectiveness like sensitivity, miss rate, precision, F1 score, etc. [93]. In this case, the weighted arithmetic mean, is the proper approach. Consequently, after a series of many experiments, the concept of comparing the results was proposed by introducing a novel and universal performance index to search for a compromise image resolution between the speed and accuracy:

$$\rho = w_\rho \cdot a \ + \left(1 - w_\rho\right) \cdot \mathrm{FPS}, \tag{41}$$

where $w_\rho \in \langle 0,1 \rangle$ weights the overall accuracy a and $(1 - w_\rho)$ weights the processing speed expressed in frames per second (FPS). By this means, it is possible to control the importance of accuracy versus FPS when designing the system. Using this performance index, it is possible to evaluate classifiers but also to select the best input resolution for a particular classifier taking the camera specificity (image resolution, camera type) into account.

Very often, during the design process, it is assumed that the processing speed is measured in FPS. It is a very important factor in the real-time processing, especially in embedded systems. It characterizes the algorithms used, the computational platform, and finally the computation costs.

However, direct use of the real FPS values makes the performance index related to the speed of the used computational platform (both hardware and software-wise). That is why, to omit this drawback, it was proposed to use the relative value $\mathrm{FPS}_{\max}/\mathrm{FPS}_{\mathrm{cal}}$, where $\mathrm{FPS}_{\mathrm{cal}}$ is the calculated value of FPS with a given resolution and $\mathrm{FPS}_{\max}$ is the

maximum value of FPS achieved over all possible resolutions. Therefore, the practical version of the proposed normalized performance index formula is as follows:

$$\rho_{\mathrm{FPS}} = w_{\rho} \cdot \frac{a}{100} + \left(1 - w_{\rho}\right) \frac{\mathrm{FPS}_{\max}}{\mathrm{FPS}_{\mathrm{cal}}} \tag{42}$$

Thus, both $a/100$ and $\mathrm{FPS}_{\max}/\mathrm{FPS}_{\mathrm{cal}}$ remain in the normalized $\langle 0,1 \rangle$ range

Moreover, an additional modified version of the performance index is proposed. It will be used to evaluate the object classification stage itself (it is used in the experiments performed in this chapter). For this purpose, the relative value $t_{\mathrm{cal}}^{-1}/t_{\min}^{-1}$ is used instead of the real FPS parameter, where t_{cal} is the mean calculation time of one test sample with a given resolution (the time for extraction of the features plus the classification time) and $t_{\min}$ is the minimum calculation time achieved over all possible resolutions. Therefore, the third version of the proposed normalized performance index dedicated to the assessment stage of the object classification formula is as follows:

$$\rho_{\mathrm{n}} = w_{\rho} \cdot \frac{a}{100} + \left(1 - w_{\rho}\right) \frac{t_{\min}}{t_{\mathrm{cal}}}. \tag{43}$$

Thus, both $a/100$ and $t_{\min}/t_{\mathrm{cal}}$ remain in the normalized $\langle 0,1 \rangle$ range.

Figure 36 shows the resulting processing scheme with the proposed procedure of tuning the pedestrian classification process using the introduced performance index. There are the same processing stages in this scheme as those in Figure 1 and Figure 3, i.e., acquisition of the IR image at the input, ROI generation, and pedestrian classification. To tune the classifier and perform tests with various image resolutions, after generating the ROI, all generated objects are resized (by upscaling or downscaling them) to many various resolutions to match with the resolution of the classifier. The following resolutions were adapted starting with 64×128 down to 16×32 in 13 steps. Then, the classification quality is measured with the proposed performance index. Finally, the best resolution of the classifier is selected for the given input data.

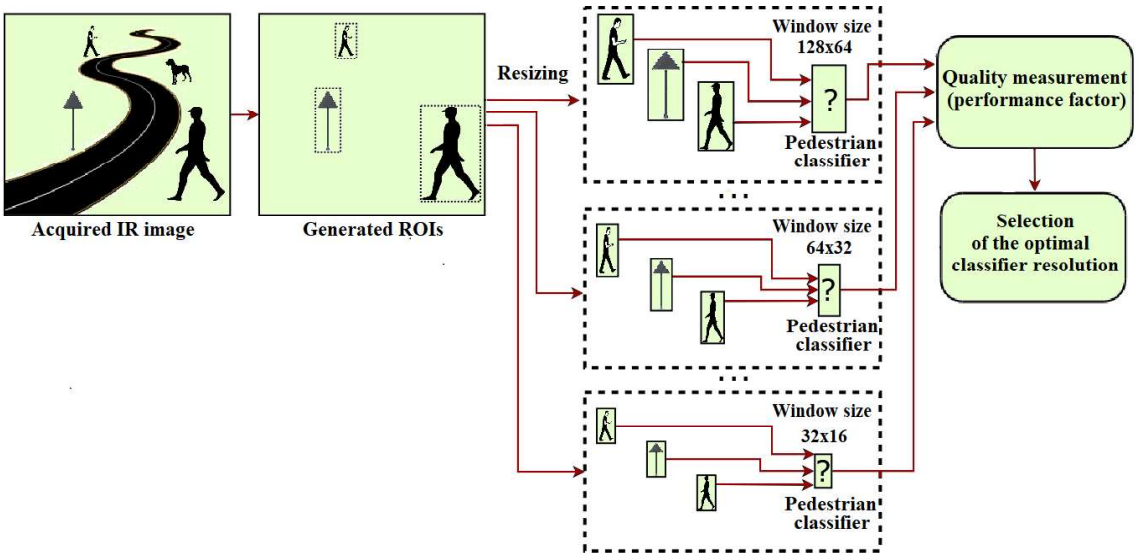


Figure 36. Processing scheme for tuning of pedestrian classification with the proposed performance index

5.2. Experiments with various input resolutions

In order to find the best resolution of the classifier applying the proposed performance index, many experiments were performed with various scenarios and using various night-vision video datasets containing pedestrians. For this purpose, a special testbed was built (Figure 37). In the experiments, particularly an impact of the image resolution was checked, classifier type, and the resulting number of features on the classification accuracy and the computation time, using three detectors, namely: HOG + SVM, ACF, and the deep CNN model.

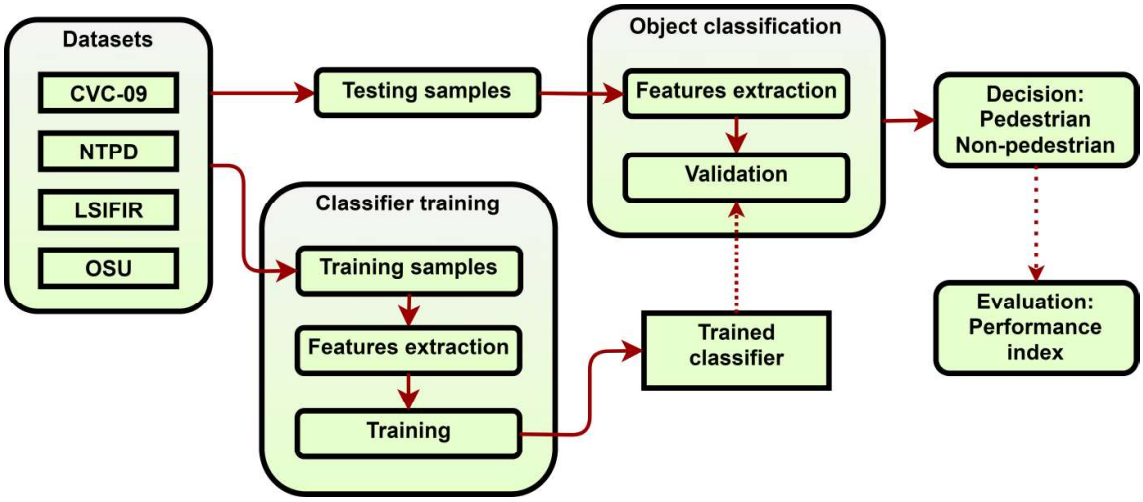


Figure 37. Testbed for comparison of tested classifiers

5.2.1. Classifier training

The numbers of training and test samples in the prepared night-vision datasets are quite varying, but statistically sufficient to conduct relevant experiments. All the prepared datasets are intentionally unbalanced as they have much more negative samples than the positive ones. This is because such relation is typical in reality for the target application (i.e., detection of pedestrians from a car at night, where images with no pedestrians occur much more often than those containing pedestrians). This however can lead to problems with the proper training of the classifier. If the classifier is trained to achieve the lowest possible learning error, this can lead to some reduction in the false-positive rate [94]. This is related to the greater number of negative slack variables that affect the objective function. To properly train the classifier with unbalanced data, in both data classes the samples should be weighed as follows

$$C_1 = w_1 C, \;\; C_2 = w_2 C \;\; \text{with } w_1 + w_2 = 1, \tag{44}$$

where C determines the importance of the misclassification and is the Lagrange multiplier upper bound, used as the penalty parameter [94].

5.2.2. Resolution of the classifier

To perform experiments with different resolutions of the classifier, the initial images were scaled into several sizes: 64×128, 56×120, 56×112, 56×104, 48×96, 40×88, 40×80, 40×72, 32×64, 24×56, 24×48, 24×40, 16×32 (7 of them are presented in Figure 38). From all of them, 13 sets of testing images were formed. These sets were prepared separately for individual datasets.

As mentioned in Section 2.2.1, the CVC-09 dataset has the pedestrians captured with many different sizes. In consequence, the initial resolutions varied a lot: from 3×6 pixels up to 190×458 pixels. To match these resolutions to the resolution of the classifiers, each image was scaled into the closest resolution of someone from the 13 listed above resolutions. Due to a relatively large span of the assumed classification resolutions, most of the images required slight scaling only.

On the other hand, in the rest of the used datasets, the initial resolutions were fixed. In the NTPD dataset, it is 64×128 pixels, whereas in the LSI FIR and OSU datasets it is 32×64 pixels (after extraction, cf., Section 2.2). It was assumed that the images were scaled down only (scaling up brings no additional information, but complicates the calculation, thus it is unreasonable). Finally, 13 test sets were created from the NTPD dataset, while 5 test sets (numbered from 9 to 13) were prepared from each of the LSI FIR and OSU datasets. The bilinear interpolation technique was used.

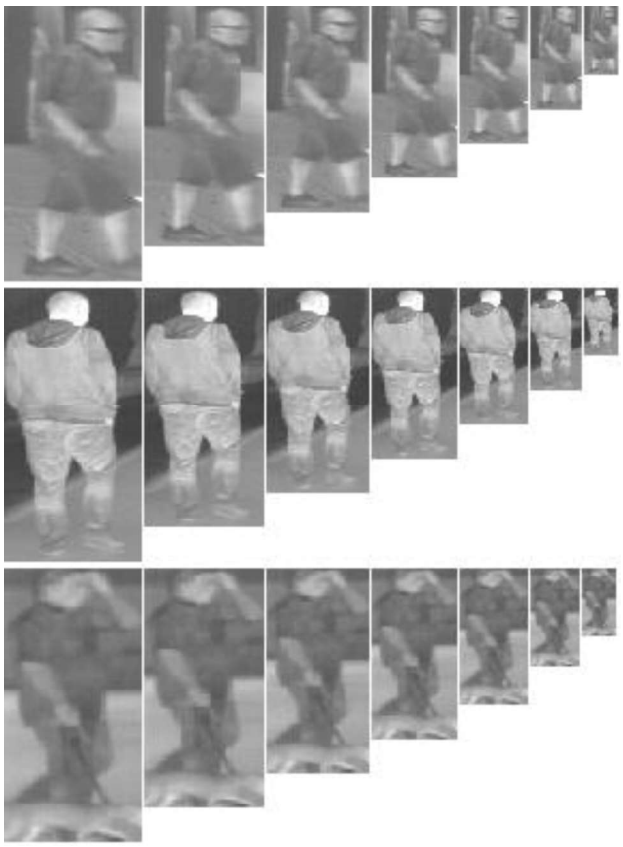


Figure 38. Three positive samples in various resolutions: 64×128, 56×112, 48×96, 40×80, 32×64, 24×48, 16×32; original images are in the CVC-09 dataset

5.2.3. Configuration of HOG+SVM and ACF detectors

While the resolution of images in different sets was varying, the rest of the parameters for the HOG feature extractor was kept constant. For all test sets the number of HOG orientation bins was set to 9, block size to 16×16 pixels, and the cell size to 8×8 pixels. For the SVM classifier, the linear kernel was used.

The ACF detector was implemented similarly as presented in [32]. In the case of night-vision and gray-scale images (both passive and active ones) the ACF was adopted to have 8 feature channels: 6 HOG orientation bins, one normalized gradient magnitude, and one luminance channel (instead of three LUV color channels used in the source solution [ref]). The AdaBoost was used as a classifier in the ACF detector to train and combine 2048 depth-two trees.

For both feature extractors, various resolutions strongly affect the number of features, which have to be analysed by the classifier (cf., Table 33).

5.2.4. Configuration of AlexNet/CaffeNet CNN

The original AlexNet/CaffeNet CNN architecture [78], [79] was prepared for images of 224×224 resolution. This network is often used for classification purposes [25], [43].

The CNNs are often used without an adaptation of the network input resolution to the resolution of the specific dataset or camera. Unfortunately, it is a common but ineffective practice, especially in the networks with a complicated structure. In the case of CNN, any change in the resolution of the CNN input layer causes the necessity of adaptation in the other layers. In consequence, it is complicated, and therefore designers try to omit it.

In this case, the CNN architecture and input resolution was adapted manually to the lowest tested resolution, i.e., to 16×32 (from resolution of 224×224) by reducing the size of the convolutional filters and the size of the maximum pooling. Then, this modified structure was used for all tested input resolutions (and only this value was changing in the structure of CNN) to ensure a fair comparison between carious resolutions. presents the details.

According to the image resolution, the number of CNN parameters is very high and varies from ca. 7 million to more than 38 million (cf., Table 33).

The datasets training sets for the training of CNNs were divided into two sets: training set (70% of images) and validation set (30% of images).

Each time the network was trained in a maximum of 25 iterations. In addition, the network training was stopped if 10 subsequent iterations did not increase validation accuracy and in such case, the configuration of the best model was restored. It was the so-called early stopping process.

Before each iteration, the training set was processed by slight random transformations to improve the generalization process. Such random transformations are often called in-place /on-the-fly data augmentation and are used to avoid network overfitting. Following techniques were used: zooming, in the range within which the random zooming to the images may be applied (value set to 0.1), and horizontal flipping.

Table 33. Number of features of HOG and ACF feature extractors, and number of parameters in the adapted CNN for various resolutions

	Number of features		Number of parameters
Input resolution [px]	**HOG**	**ACF**	**CNN**
64×128	3780	4096	38 686 369
56×120	3024	3360	32 657 057
56×112	2808	3136	30 822 049
56×104	2592	2912	28 987 041
48×96	1980	2304	24 006 305
40×88	1440	1760	19 549 857
40×80	1296	1600	18 239 137
40×72	1152	1440	16 928 417
32×64	756	1024	13 520 545
24×56	432	672	10 636 961
24×48	360	576	9 850 529
24×40	288	480	9 064 097
16×32	108	256	7 229 089

Table 34. Proposed CNN structure

Layer number	Layer type	Elements	Activation function	Remarks
1	convolutional	48, 7×7 filters	ReLU	maximum pooling, filter size 2×2, local response normalization
2	convolutional	128, 5×5 filters	ReLU	maximum pooling, filter size 2×2, local response normalization
	convolutional	192, 3×3 filters	ReLU	-
	convolutional	192, 3×3 filters	ReLU	-
	convolutional	128, 3×3 filters	ReLU	maximum pooling, filter size 2×2
6	fully connected	2048 neurons	ReLU	dropout ratio of 0.5
7	fully connected	2048 neurons	ReLU	dropout ratio of 0.5
8	output	1 neuron	sigmoid	pedestrian detection score

5.2.5. Classification accuracy and calculation time

At the beginning of the tests, the CVC-09 dataset was used, because, it presents a very similar material to that occurring in real situations. The images were taken during the day and at night. The pedestrian regions have various sizes and therefore the analysed ROIs have various resolutions. In the next step, tests were performed with the NTPD, LSI FIR, and the OSU datasets.

The obtained results are described in detail below, listed in Table 35 (which has the following columns: dataset, set name, frame size, classification accuracy, and calculation time). and are presented as graphs in Figure 39, Figure 40, and Figure 41. For each test set, the classification accuracy was calculated and the mean calculation time.

The calculated classification accuracy values constitute points with equal false and miss detection probabilities. These points were computed with 180 test samples (90 positive and 90 negative).

The determined mean calculation times are composed of two phases: duration of the feature extraction process and time needed for the classification of a single test sample. The processing was implemented in the C# programming language with EmguCV v. 2.4.10 environment [95] and LIBSVM [96] as the SVM library. The CNN was implemented with Keras and TensorFlow [97] using Python language. The training process was performed with the GPGPU support in the Google Colab cloud environment. The usage of GPU allows parallelization of processing and therefore substantial speed-up of processing, but it strongly depends on many factors like the algorithm and data structure, or architecture of the GPU. Therefore, in this dissertation, the computations during the classification stage were made with a single CPU core to make fair, hardware-independent comparisons between various methods and image resolutions. The following hardware was used: CPU Intel Core i7-6950X, 8 GB of RAM.

5.2.6. Discussion of results

The best classification accuracy was achieved with the CNN approach, but results obtained by other classifiers are also fully acceptable (cf., Table 35 and Figure 39, Figure 40 and Figure 41). In Table 35 the results of classification accuracy and calculation time are highlighted, which are the best in the set of various resolutions of a given dataset and those which are close to the maximum values but obtained with lower resolutions. It should be noticed, that in almost all cases (especially for the CNN) the results are good even for low-resolution input data.

For example, for the resolution of 24×40, the accuracy is almost as high (99.89%) as for the highest resolution among all datasets. Furthermore, for the CVC-09 daytime and NTPD datasets, the best accuracy is obtained for a lower resolution (40×72) than the maximum 64×128. The right columns of Figure 39, Figure 40, and Figure 41 show that for the CNN detector the graphs of the classification accuracy are almost flat.

The resolution of a sample strongly affects the processing time. It is true for all the classifiers. The CNN is the slowest solution (more than 20 times slower than the HOG+SVM or the ACF detector). For low-resolution samples (e.g., 16×32) it needs ca. 5.5 ms and for high-resolution ones (e.g., 64×128), it needs ca. 25 ms to calculate the result. The ACF detector is slightly slower than HOG+SVM, but achieves higher accuracy, especially for the CVC-09 and NTPD datasets. For processing low-resolution samples (e.g., 16×32), the HOG+SVM detector needs 0.08 ms only, while ACF needs 0.21 ms. For high-resolution samples (e.g., 64×128), the HOG+SVM needs about 0.75 ms to calculate the result while the ACF needs 1.15 ms (cf., Table 35).

Table 35. Classification accuracy and calculation time for various resolutions and classifiers

Dataset	Set	Frame Size [px]	Classification accuracy (*) [%]			Calculation time (**) [ms]		
			HOG+SVM	ACF	CNN	HOG+SVM	ACF	CNN
CVC-09 day-time subset	1	64×128	92.9	**98.12**	**99.56**	0.74	1.17	24.41
	2	56×120	93.4	97.24	99.20	0.59	0.99	20.79
	3	56×112	93.5	96.83	99.38	0.57	0.93	19.70
	4	56×104	93.7	96.72	99.32	0.52	0.85	18.48
	5	48×96	93.6	96.88	99.12	0.49	0.79	15.53
	6	40×88	**94.2**	96.55	99.24	0.34	0.59	13.05
	7	40×80	**94.0**	96.43	99.21	0.30	0.51	12.32
	8	40×72	93.8	96.18	**99.34**	0.27	0.45	11.35
	9	32×64	93.8	95.83	98.83	0.21	0.40	9.25
	10	24×56	93.1	94.34	98.92	0.15	0.32	7.71
	11	24×48	92.9	94.48	98.75	0.13	0.28	7.39
	12	24×40	92.3	93.89	98.93	0.11	0.26	6.83
	13	16×32	90.7	91.83	98.34	**0.08**	**0.23**	**5.23**
CVC-09 night-time subset	1	64×128	**96.6**	**98.53**	98.28	0.73	1.15	24.60
	2	56×120	95.5	97.77	**98.71**	0.59	0.95	20.62
	3	56×112	95.3	97.75	98.07	0.56	0.93	19.63
	4	56×104	95.5	**97.50**	98.61	0.55	0.84	18.54
	5	48×96	94.8	97.14	98.43	0.40	0.79	15.54
	6	40×88	94.7	96.67	98.31	0.36	0.59	12.97
	7	40×80	94.4	96.72	98.26	0.29	0.52	12.26
	8	40×72	94.2	96.48	98.59	0.27	0.45	11.25
	9	32×64	93.3	96.34	98.14	0.21	0.39	9.34
	10	24×56	93.2	95.38	98.42	0.14	0.30	7.72
	11	24×48	92.5	94.64	98.15	0.13	0.29	7.42
	12	24×40	92.2	93.82	**98.48**	0.11	0.25	6.88
	13	16×32	89.4	91.67	97.85	**0.08**	**0.23**	**5.46**
NTPD	1	64×128	**98.94**	98.69	**99.23**	0.76	1.14	27.37
	2	56×120	98.78	98.70	99.16	0.60	0.98	20.53
	3	56×112	98.61	98.71	99.14	0.55	0.89	19.70
	4	56×104	98.56	98.74	98.98	0.55	0.84	18.46
	5	48×96	98.57	98.85	98.99	0.43	0.79	15.55
	6	40×88	98.74	99.03	98.99	0.34	0.61	12.99
	7	40×80	**98.91**	**99.03**	98.96	0.31	0.52	12.29
	8	40×72	98.78	98.98	99.26	0.28	0.44	11.32
	9	32×64	98.34	98.61	98.92	0.22	0.39	9.58
	10	24×56	97.77	98.02	98.61	0.16	0.32	7.70
	11	24×48	97.65	97.43	98.81	0.18	0.29	7.50
	12	24×40	97.25	97.21	**98.94**	0.14	0.23	6.93
	13	16×32	95.02	94.26	98.48	**0.09**	**0.21**	**5.50**
LSI FIR	9	32×64	98.74	**99.33**	**99.47**	0.22	0.37	9.50
	10	24×56	**99.01**	98.96	99.33	0.19	0.35	7.75
	11	24×48	98.72	98.82	99.33	0.17	0.29	7.44
	12	24×40	98.31	98.64	99.45	0.13	0.27	6.87
	13	16×32	96.58	97.04	**99.41**	**0.10**	**0.23**	**5.48**
OSU	9	32×64	**99.79**	99.87	99.77	0.22	0.40	9.24
	10	24×56	99.58	**99.90**	99.93	0.19	0.32	7.69
	11	24×48	99.65	99.31	**99.96**	0.18	0.31	7.45
	12	24×40	**99.27**	98.83	99.89	0.13	0.25	6.86
	13	16×32	95.03	97.81	98.87	**0.09**	**0.24**	**5.53**

(*) The classification accuracy is a point on the DET curve with equal false alarm miss probabilities. (**) The presented mean calculation time takes a sum of the process of features extraction and classification of one test sample mean times into account.

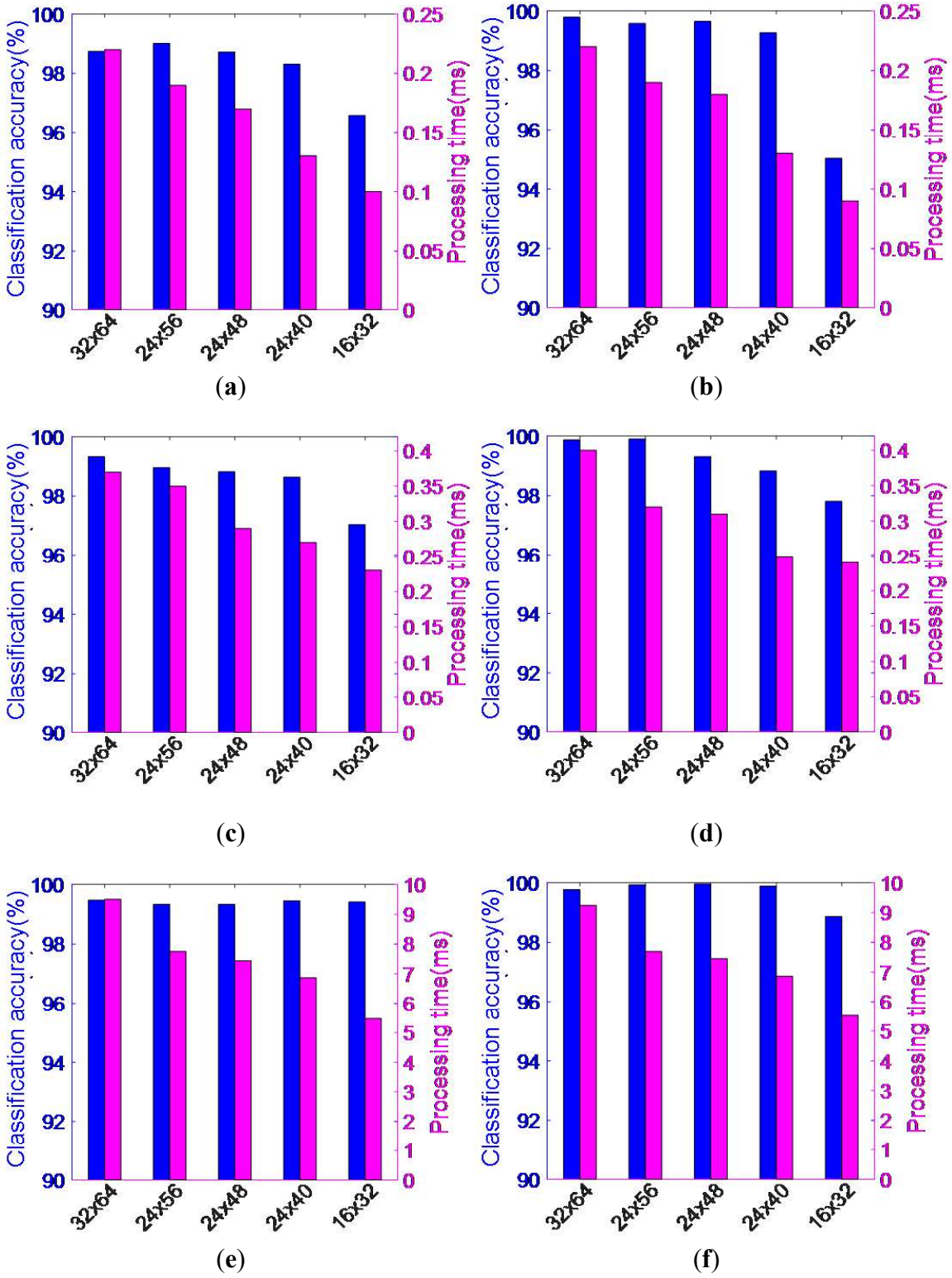


Figure 39. Classification accuracy and processing time as functions of image resolution: HOG + SVM classifier (first row), ACF detector (second row), CNN (third row) for the following datasets: LSIFIR (first column: a–c), OSU (second column: d–f)

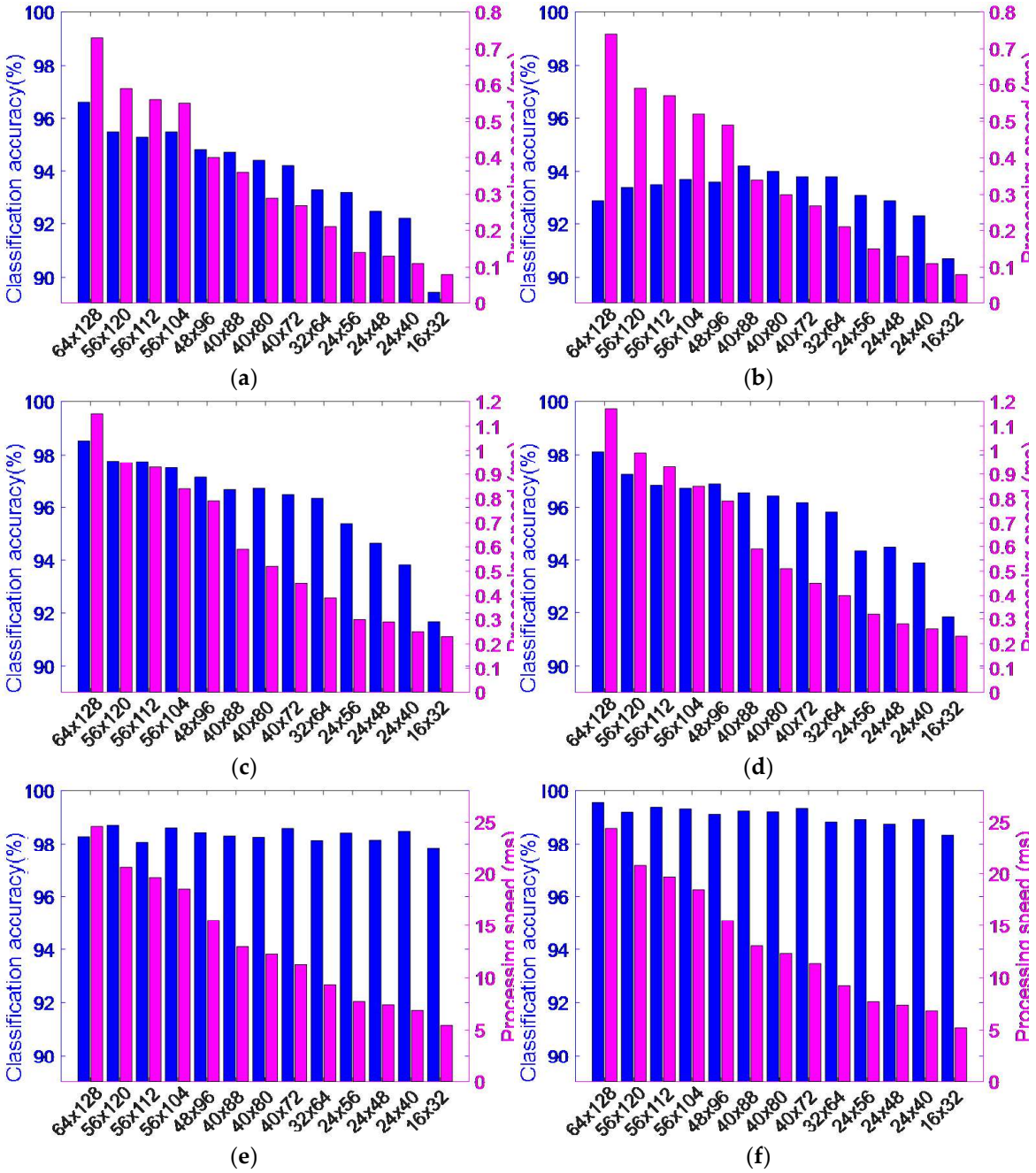


Figure 40. Classification accuracy and processing time as functions of image resolution: HOG+SVM classifier (first row), ACF detector (second row), CNN (third row) for the following datasets: CVC-09 night-time (first column: a, c, e), CVC-09 day-time (second column: b, d, f)

It could be noticed that in the case of the CVC-09 datasets, the obtained classification accuracy values are very good (above 90%) for all tested detectors (cf., Table 35, Figure 39, Figure 40 and Figure 41). It can also be seen that in the day-time subset of the CVC-09, high effectiveness can be achieved with the HOG+SVM detector with a relatively low resolution of samples, i.e., just 40×88. The ACF detector achieves local optima with the resolution of 48×96. For the night-time subset of the CVC-09 dataset, both detectors (i.e., SVM and ACF) achieve mild local maxima of the effectiveness with the resolution of samples equal to 56×104 (cf., Set 4 in Table 35 and Figure 40). In the night-time sets the detectors achieve better results than those for the

day-time sets. It is due to the fact, that the thermal contrast at night is higher than on a day (cf., Figure 7). During the analysis of other datasets (i.e., NTPD, LS IFIR, and OSU) the values of the obtained detection effectiveness are better than those for the CVC-09 dataset (all of them are above 95%, in many cases larger than 98%). It is valid for all the resolutions (even very low) and all the classifiers. For the LSI FIR and OSU datasets the classification with the resolution equal to 24×48 achieves similar accuracies to the best ones but with approximately 20% shorter time than this for the initial resolution (cf., Figure 39). For the NTPD dataset, the classifier resolution can be reduced to 40×80 while the effectiveness remains almost unchanged. By this reduction, the classification time is shorter by approximately 60% (cf., Figure 41).

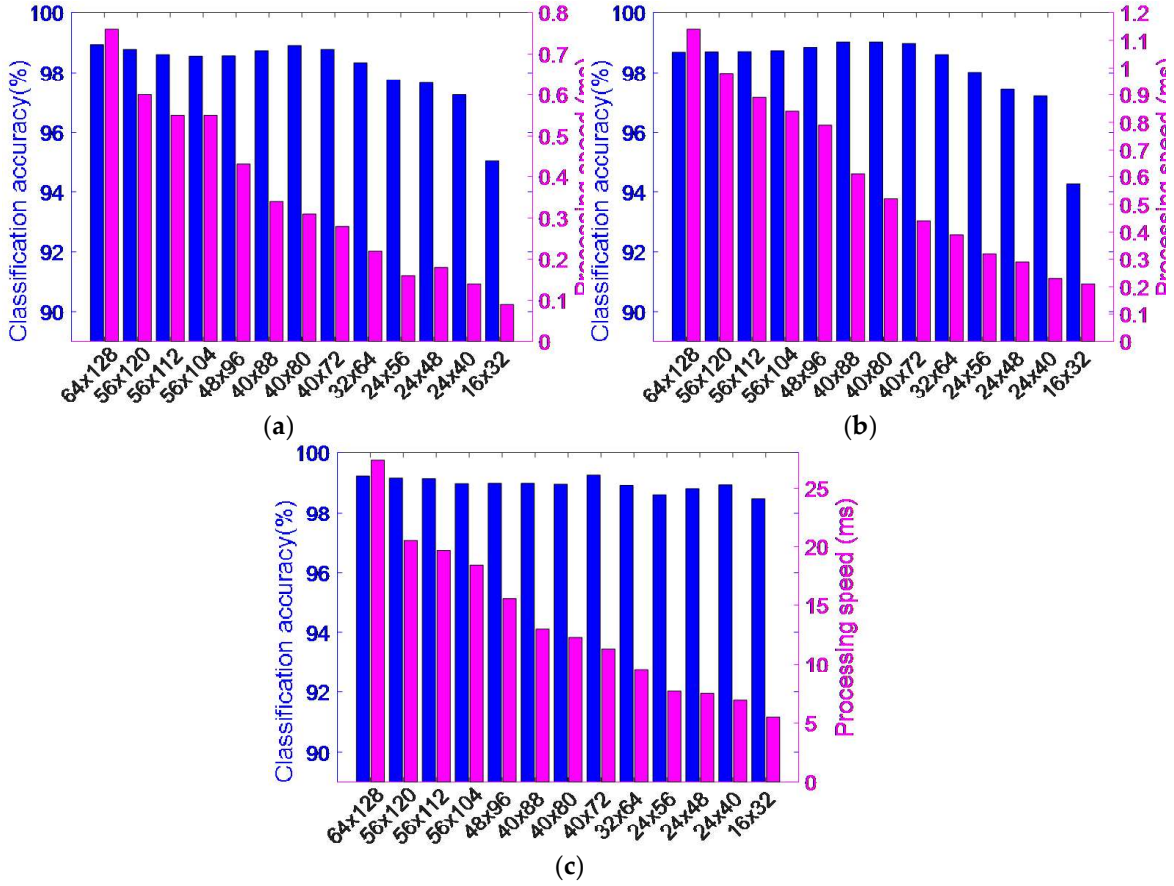


Figure 41. Detection rate and processing time as functions of image resolution for NTPD dataset: HOG+SVM classifier (left), ACF detector (middle), CNN (right)

Concluding, the classification effectiveness does not diminish significantly, even if the image resolution substantially decreases. The upper limit of the classifier error is related to the dimension of the features vector and the number of the training samples [94]. This relation is visible in the experiments (cf., Table 35, Figure 39, Figure 40, and Figure 41). Thus, it can be stated that, in general, the resolution of the classifier can be lower than the original resolution of the analyzed images. However, the best resolution should be chosen with the use of the proposed performance index.

5.3. Performance index results

Using the results of experiments (Table 35) and equation (42), the performance indices η were calculated for all datasets, resolutions, and tested classifiers. The results are presented in Figure 42, where values in the x-axis refer to the testing sets presented in Table 35 (a continuous type of chart was used for better readability, despite the discontinuous x-axis domain).

As already mentioned in Section 5.1, the weighted sum of the relative accuracy $a/100$ and the inverse of the relative time $t_{\min}/t_{\mathrm{c}}$ is the proper approach to define the appropriate performance index ρ for experiments with an object classification step.

It could be noticed from the experiments that accuracy (a in formula (20)) is greater than 90% for almost all configurations (cf., Table 35), whereas calculation time ($t_{\mathrm{c}}^{-1}/t_{\min}^{-1}$) varies in a large extent. Taking into account the type of the considered detection system, i.e., the pedestrian detection, thus, the variation of accuracy should have a significantly higher influence on the performance index than a variation of the mean calculation time.

Therefore, three values of the weight w_{ρ} were proposed for the performance index, depending on the application. These values were selected experimentally and adjusted as closely as possible to the three proposed application scenarios.

In the first scenario, where the processing time is assumed to be very important, e.g., in applications with low power processing units like vehicles, the weight should be set to ca. $w_{\rho} = 0.92$ (Figure 42a,b). In result, the performance index is higher for low object resolutions.

In the second scenario, where the accuracy is assumed to be much more important, e.g., for offline processing of CCTV recordings or safety and security systems, the weight w_{ρ} should be set ca. to $w_{\rho} = 0.98$ (Figure 42e,f). In result, the performance index achieves the highest values for medium and high resolutions of the classifiers.

In the third scenario, in case of the balanced configuration, still with high accuracy importance, and taking changes in the processing time into account, e.g., in automotive and real-time security systems, the weight w_{ρ} should be set ca. to $w_{\rho} = 0.95$ (Figure 42c,d).

Most curves in Figure 42 have global and local maxima. They were selected to state the best performance resolutions for the tested classifiers. The results are collected in Table 36. Besides the best resolution, differences in accuracies and processing times are presented (in percent), in relation to the classifier with the highest resolution. The difference in accuracy varies from -2.22% to 0.97%, as the reduction of the processing time reaches up to 74% (cf., Table 36).

For some cases (as presented in Table 36), both the time reduction and the increase of the classification accuracy could be achieved (by means of the resolution reduction). Classifiers, which are tuned for the best performance index can process data up to four times faster than non-tuned classifiers with a slight decrease of the accuracy (merely by about 1−2%).

There is no universal best resolution for all cases, but the best performances are achieved for resolutions between 24×40 and 48×96 pixels (cf., Table 35, testing sets from No. 7 to 12).

With the proposed performance index, the best input resolution can be effectively selected in a given dataset and classifier. With such newly reduced resolution, which is typically much lower than the initial resolution (that of the input images), the processing time needed for the classification could decrease by up to 74% (percentage difference referred to the classifier with the highest resolution) with insignificant influence on the accuracy.

Moreover, the presented approach is quite general, i.e., it is applicable not only to the considered problem but also to the detection of any type of object with any classifier.

Table 36. Configuration sets, classification accuracy and processing time for testing subsets

Dataset	**Type of classifier**	**Best performance resolution**	**Difference in accuracy (*) [%]**	**Processing time reduction (*) [%]**
LSIFIR	SVM	24×56	+0.27	−13.64
	ACF	24×40	−0.69	−65.56
	CNN	16×32	−0.06	−42.31
OSU	SVM	24×48	−0.14	−18.18
	ACF	24×56	+0.03	−13.64
	CNN	24×40	+0.12	−25.76
NTPD	SVM	40×72	−0.16	−63.16
	ACF	40×72	+0.29	−61.41
	CNN	40×72	+0.03	−58.64
CVC-09 Day-time	SVM	32×64	+0.97	−71.62
	ACF	48×96	−1.06	−33.33
	CNN	24×40	−0.63	−72.02
CVC-09 Night-time	SVM	40×80	−2.28	−60.27
	ACF	32×64	−2.22	−66.09
	CNN	24×40	+0.21	−73.83

(*) Percentage difference referred to the classifier with the highest resolution.

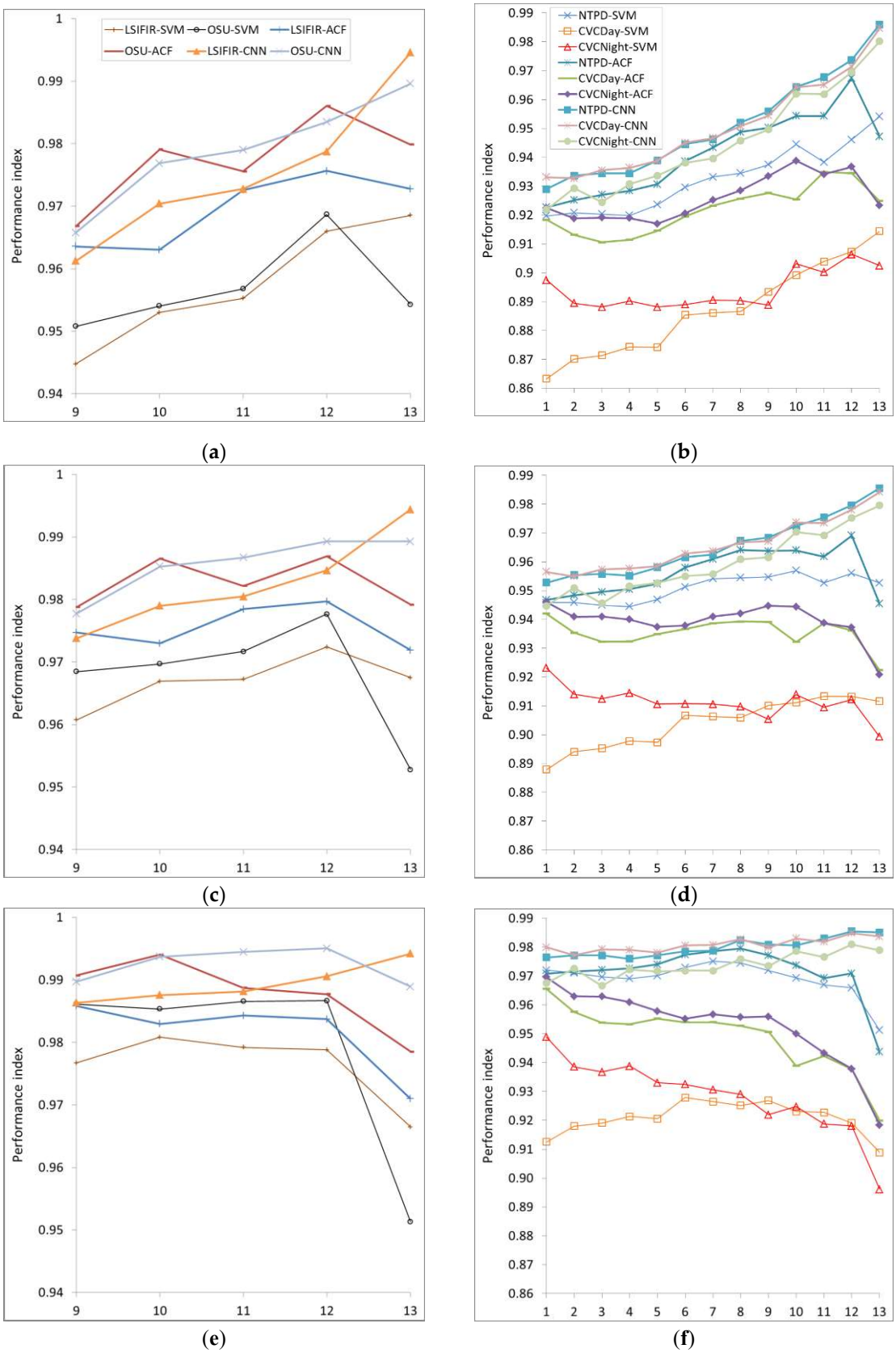


Figure 42. Performance indices as functions of image resolutions (values on x-axis refer to particular test sets in Table 35): for (a,b) $w_\rho = 0.92$, for (c,d) $w_\rho = 0.95$, for (e,f) $w_\rho = 0.98$, with w_ρ being the weight of accuracy for various datasets and classifiers indicated by various colors as explained in the legend (the first legend is common for a), c), e) plots, and the second one for b), d), f) plots)

6. Experiments with the proposed pedestrian detection procedure

The general procedure of the proposed detection process with the introduced improvements is presented in the diagram in Figure 3. These improvements increase the computational efficiency and accuracy of pedestrian detection. More detailed diagrams presenting individual stages are presented in the previous chapters: the ROI generation procedure is presented in Figure 20 (in Chapter 3), the idea of ROI adjustment is presented in Figure 35 (Chapter 4), and the procedure of tuning the pedestrian classification with the proposed performance index is presented in Figure 36 (Chapter 5).

This chapter presents the experiments performed to evaluate the impact of the proposed improvements on the accuracy and computational efficiency of the entire pedestrian detection process.

Two baseline, commonly used detectors were used for this task, namely the ACF [76] and CNN / AlexNet [78], [79] (details of settings and implementation are described in Sections 5.2.3 and 5.2.4). These detectors are well suited for carrying out a large number of experiments and are relatively easy to adapt to different input resolutions. The adaptation is much more difficult with complex CNNs, where changing the input resolution necessitates adjusting multiple layers of the network. It is difficult to compare the results of such differently modified CNNs with each other (for various input resolutions). In addition, the ACF and AlexNet detectors are well described in the literature, which allows the results to be compared.

The tests were conducted on two datasets: CVC-14 and KAIST, which contain thermal images recorded at night with annotated test sequences that allow to perform experiments with pedestrian detection algorithm (both datasets are described in Section 2.2).

The first section of this chapter presents the description of the implementation of the proposed algorithm created by the author of this dissertation. The following sections present the initial experiments for the settings obtained with the proposed ROI generation technique, the experiments with selecting the classifier resolution with the proposed performance index, and experiments with adjustment of segmented ROIs. Then, the obtained results are compared with some other pedestrian detection methods, i.e. those based on segmentation with the sliding window and some selected from the literature, in which similar detectors were used, i.e., ACF and CNN/AlexNet.

6.1. Implementation

For the experiments, the proposed pedestrian detection algorithm was implemented along with the tools that enabled the assessment of the proposed improvements. The implementation was made in C# in the Visual Studio environment. The current implementation uses the multicore CPU and can run in a multi-threaded architecture (as presented in Figure 43).

The software has an object-oriented, modular architecture that allows for easy attaching of new datasets, new ROI generation approaches, feature extractors, and a variety of classifiers. The software includes the main proposed pedestrian detection algorithm (divided into classes) and a demonstration application named ThermalPDdemo (see Figure 44). This application is built as a WindowsForm application and enables to:

- set paths to datasets,
- train/load ACF detector parameters,
- load pre-trained CNN model with defined structure,
- perform frame by frame detection with ACF detector or CNN,
- evaluate pedestrian detection algorithm on entire dataset of images.

The software is based on the few main abstract classes: *TDataset*, *TFeatureExtractor*, *TClassifier*, *TSegmentator*, which constitute an interface for the implementation of individual algorithms. In addition, the software has the class *PedestrianDetectionModule*, which is the main module for functions used inside the pedestrian detection process.

The general structure of implemented software (most important classes) is as follows:

- *ThermalPDdemo* – window application (presented in Figure 44),
- *PedestrianDetectionModule* – main pedestrian detection module,
- *TSegmentator*,
  - *Otsu_TSegmentator*,
  - *Fixed_TSegmentator*,
- *TDataset*,
  - *CVC-14_TDataset*,
  - *KAIST_TDataset*,
- *TFeatureExtractor*,
  - *ACF_FeatureExtractor*,
  - *CNN_FeatureExtractor*,
  - *HOG_FeatureExtractor*,
- *TClassifier*,
  - *SVM_Classifier*,
  - *Boost_Classifier*,
  - *CNN_Classifier*.

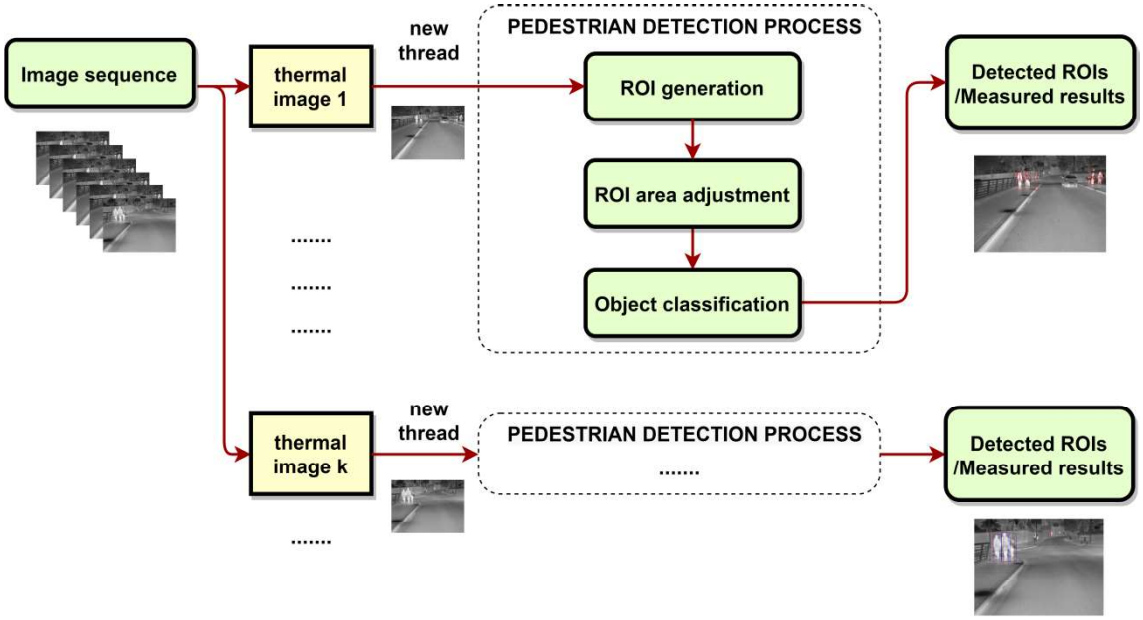


Figure 43. Block diagram of the multi-threaded implementation of the proposed pedestrian detection algorithm

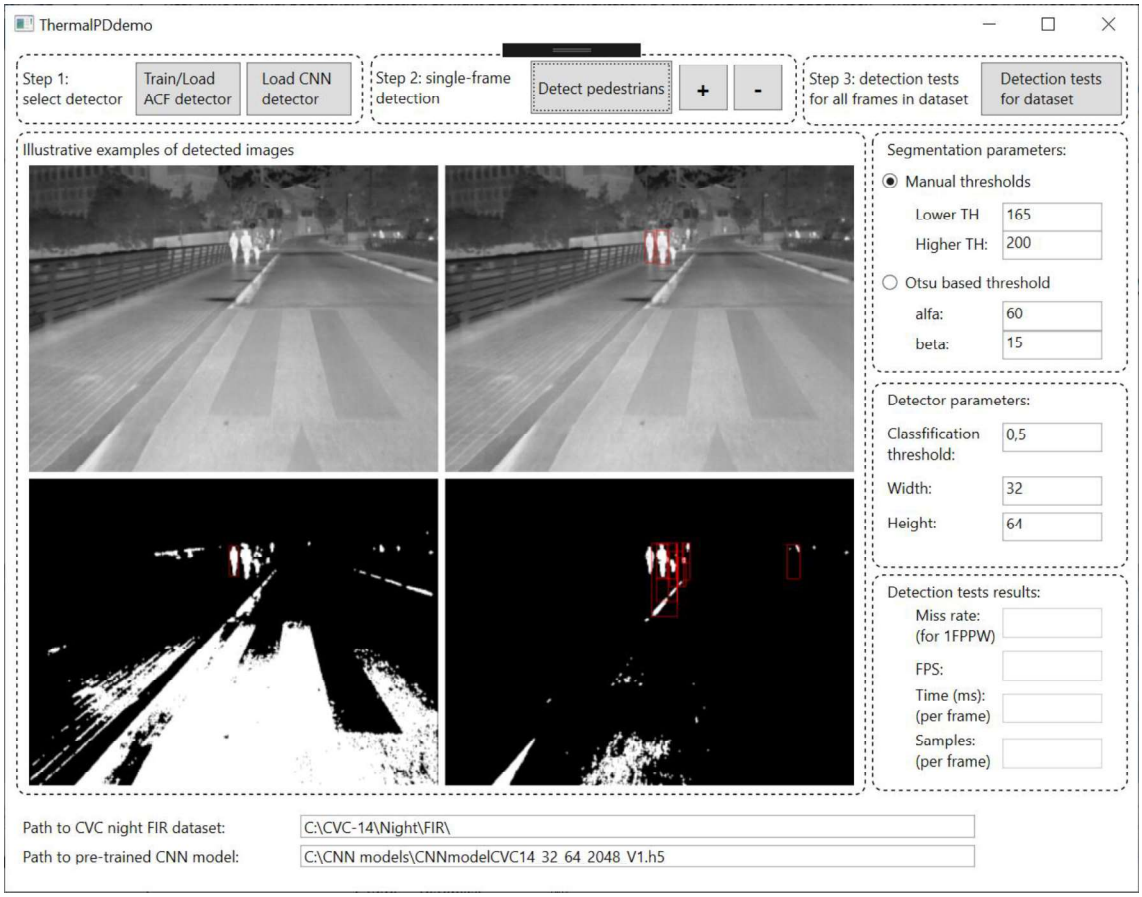


Figure 44. Window of application *ThermalPDdemo* for conducting experiments with the pedestrian detection algorithm

The software uses several external libraries, including EmguCV (C # version of OpenCV library), Keras, and TensorFlow (to support CNNs) [97]. Libraries are installed into a project via NuGet packages in Visual Studio (Project -> manage NuGet packages) [98]. The list of all NuGet packages used in the project is as follows:

- VDK.EmguCV.x64 – v. 2.4.10,
- TensorFlow.NET – v. 0.14.0,
- Keras.NET – v. 3.6.2.4,
- Numpy.Bare – v. 3.7.1.20,
- libsvm.net – v. 2.1.7,
- IKVM.OpenJDK.Core – v. 7.2.4630.5,
- IKVM.OpenJDK.Util – v. 7.2.4630.5.

6.2. Experiments

This section presents the experiments performed to verify the accuracy of the proposed pedestrian detection procedure. The following subsections include the initial experiments for the settings obtained with the proposed ROI generation technique, experiments with selecting the classifier resolution with the proposed performance index, and experiments with adjustment of segmented ROIs.

6.2.1. Methodology

The main metrics used in this chapter are MR, FPPI, and FPS, which are described in Section 3.8.1.

The accuracy of the proposed pedestrian detection algorithm is assessed based on the relation of MR to FPPI. These metrics are closely related to each other, as the MR value decreases, the FPPI value increases and vice versa. On the one hand, it is important that the MR should be as small as possible to detect pedestrians with a high accuracy. On the other hand, it is also important that the FPPI value should also be low to ensure proper operation of the system and to avoid frequent false detections. Presented MR values in the experiments were achieved for the mean FPPI value equal to 1. However, in the experiments also detection error trade-off (DET) curves (plots of MR values depending on the FPPI values) will be presented.

During experiments, it was noticed that for both the CVC-14 and the KAIST datasets, the pedestrians in the far distance from the camera were not always annotated by the authors of the datasets. The pedestrians that are visible from a long distance but not annotated become annotated on successive frames of the sequence as they were approaching the camera. However, the proposed pedestrian detection algorithm very often detected such not annotated pedestrians at long distances. Since pedestrians were not annotated, the software comparing results with the ground-truth description, classified the detected pedestrians as false-positives (but, in fact the pedestrians were detected correctly). As a result, the average number of FPPI was even doubled (verified for the CVC-14 dataset and set 09 from the KAIST dataset), and it was necessary to increase the classification threshold to reduce the FPPI value (to achieve a value equal to 1) at the expense of increasing the MR value.

Therefore, two more precise objective metrics were proposed: MR_{far} and MR_{near} for pedestrians at distances up to 150 m and 75 m respectively.

CVC-14 and KAIST datasets differ in resolution and in the field of view. For both of them, minimum pedestrian heights (in pixels in the image) have been estimated. These values correspond to the boundary distances (150 m from the camera for MR_{far} and 75 m from the camera for MR_{near}). The estimated values of minimum pedestrian heights are presented in Table 37.

The MR_{far} metric is used to limit detection of very distant pedestrians who often are not annotated and to avoid incorrectly increasing the FPPI value. Moreover, considering that the resolutions offered by the thermal imaging cameras are relatively low (320×256 for the KAIST dataset and 640×470 for the CVC-14 dataset), it makes no need to detect and classify pedestrians at very far distances. The sizes of such pedestrians in the image are very low, e.g., even less than 8×16 pixels. For such small resolutions, the correct classification is difficult even for a human.

As a result, the annotated pedestrians outside the MR_{far} metric range are also not included in the analysis. This metric achieves stable results and better reflects the overall performance of the tested detector. Therefore, subsequent tests of the pedestrian detection procedure with the proposed improvements were conducted in relation to the MR_{far} and MR_{near} metrics only.

The MR_{near} metric determines the accuracy of pedestrian detection close to the camera (up to 75 m). For this range, pedestrians may be on a collision course with a vehicle. Therefore, it is very important that the value of MR_{near} should be as low as possible.

Table 37. Minimum pedestrian heights (in pixels) estimated for the proposed metrics for KAIST and CVC-14 datasets

	CVC − 14		**KAIST**	
Distance	Minimum height (pixels)	Included pedestrians (*) [%]	Minimum height (pixels)	Included pedestrians (*) [%]
Far (MR_{far})	50	80.3	20	76.9
Near (MR_{near})	100	37.7	40	35.4

(*) Percentage value of annotated pedestrians included within the metric range

As shown in Table 37, 80.3% of annotated pedestrians from the whole CVC-14 dataset and 76.9% from the KAIST dataset are within the metric range of MR_{far} (up to 150 m). For the MR_{near} metric (up to 75 m), there are 37.4% of all annotated pedestrians included in the CVC-14 dataset and 35.4% in the KAIST dataset.

6.2.2. Initial tests

This section presents initial experiments with the proposed pedestrian detection procedure that were conducted on the CVC-14 and KAIST datasets. Tests were

performed with settings of the ROI generation process proposed in Chapter 3 and using two baseline object classifiers ACF and AlexNet / CNN.

The tests for both detectors were performed for 32×64 input resolution. Initial and subsequent experiments for the KAIST dataset were conducted on a representative 09 subset (campus). The results are shown in Table 38.

The lowest MR_{far} values were achieved with the *balanced* settings and double thresholding for the CVC-14 dataset (MR_{far} = 24.0%) and with *balanced* settings and triple thresholding for the KAIST dataset (MR_{far} = 26.2%).

In the case of the MR_{near} parameter, the achieved values are even lower than for the parameter MR_{far}. It also confirms that the accuracy of pedestrian detection increases as the distance to the camera decreases.

In most cases, the CNN and ACF detectors achieved similar results. The biggest difference was for the CVC-14 dataset with double thresholding and *balanced* settings (for ACF, the value of MR_{far} was equal to 29.1%, for CNN the value of MR_{far} was equal to 24.0%).

Although the MR values of the proposed ROI generation step were very low (see Section 3.9), at the object classification stage, the ACF and CNN detectors make additional errors. If in a hypothetical situation, a classifier could classify the samples without an error, the resulting MR values would be very low, equal to those obtained after ROI generation stage (as presented in Chapter 3).

The relationship between MR and FPPI parameters is crucial to the operation of the pedestrian detection system. The lower the value of the FPPI parameter is required, the higher values of the MR, MR_{far} and MR_{near}.parameters, will be achieved.

Table 38. MR_{far}, MR_{near} and FPS values obtained after the initial pedestrian detection experiments with CVC-14 and KAIST datasets

Dataset	Parameter	**Double thresholding**				**Triple thresholding**			
		balanced		*best accuracy*		*balanced*		*best accuracy*	
		ACF	CNN	ACF	CNN	ACF	CNN	ACF	CNN
CVC-14	MR_{far} [%]	29.1	24.0	31.8	31.2	34.2	31.2	32.2	41.5
	MR_{near} [%]	17.1	17.9	18.1	20.3	23.5	18.9	20.2	30.3
	FPS*	27.0	1.3	17.3	0.8	17.7	0.8	9.4	0.5
	FPS**	91.0	4.5	59.6	2.3	60.4	2.3	32.1	1.5
KAIST	MR_{far} [%]	30.9	28.4	31.1	31.7	27.5	26.2	31.5	31.4
	MR_{near} [%]	28.1	23.9	23.9	23.6	20.1	18.9	23.1	24.6
	FPS*	37.3	3.5	32.1	2.3	32.1	2.8	29.0	1.8
	FPS**	143.3	11.5	123.3	7.8	118.7	9.5	112.1	5.8

(*) The FPS was calculated for a single-core of Intel Core i7-870 CPU

(**) The FPS was calculated for a four-core of Intel Core i7-870 CPU

In almost all cases, the achieved MR_{far} values are lower for *balanced* settings than for *best accuracy* settings. This is due fact that a much larger number of ROIs are obtained from the ROI generation stage with the *best accuracy* settings. For these settings, the ROI generation process is more accurate (lower MR values are achieved, for the details see Section 3.9), but due to the larger number of generated ROIs, the classifier makes more mistakes (false-positives detections), which increases the FPPI value.

6.2.3. Selection of classifier resolution

In this section, the experiments were carried out with the proposed performance index ρ (according to the formula (42)) in order to select the best input resolutions for ACF and CNN detectors for the CVC-14 and KAIST datasets. The experiments were performed for various resolutions, and the performance index was calculated taking into account the effectiveness of the entire pedestrian detection algorithm (FPS parameter). Complementary research for the object classification stage itself and in a broader context (for both, NIR and FIR datasets) was carried out in the previous Chapter 5,.

To perform experiments with various resolutions of the classifier, the ROIs were scaled into 13 sizes: 64×128, 56×120, 56×112, 56×104, 48×96, 40×88, 40×80, 40×72, 32×64, 24×56, 24×48, 24×40, 16×32 (similarly as in Chapter 5).

In both datasets, the pedestrians were captured with many different sizes. In consequence, the initial resolutions of ROIs varied a lot. To match these resolutions to the resolution of the classifiers, each ROI was scaled each time into the resolution of the tested classifier.

For all of these 13 resolutions for the CVC-14 dataset, the ACF and CNN detectors were trained and full detection tests were performed. In case of the KAIST dataset, the detectors were trained only for the 5 lowest resolutions due to the low resolution of the dataset (320×256 pixels). Experiments for both datasets were performed with double thresholding and *balanced* settings.

The results of the experiments for various input resolutions of tested detectors are presented in Table 39 and in Figure 45. Moreover, the set of finally selected resolutions based on the performance index for both datasets are presented in Table 40.

In the case of the CVC-14 dataset, the lowest MR_{far} value was achieved for both detectors with a resolution of 32×64 (29.1% for ACF and 24% for CNN). Furthermore, the highest performance index value was also obtained for this resolution (cf. Figure 45). As a result, the choice of the resolution of 32×64 allowed to almost double the efficiency of the pedestrian detection algorithm (the FPS parameter increased by 26.1% for the ACF detector and by 333.3% for the CNN) compared to the detector with the highest tested resolution.

For the KAIST dataset, the lowest obtained values of MR_{far} for the ACF and CNN detectors were similar: 29.6% for the ACF detector with a resolution of 24×56, and 28.4% for the CNN detector with a resolution of 32×64. However, the highest values of the performance index parameter (see Figure 45) were obtained for much lower resolutions for the ACF and CNN detectors, respectively: 16×32 and 24×40. With these

resolutions, the effectiveness of the entire pedestrian detection algorithm increased by 89.3% with the ACF detector and by 85.7% with the CNN detector compared to the classifier with the highest input resolution (see Table 40).

The results show that the detectors achieve good detection accuracy in all cases even for relatively low resolutions, for which the computational efficiency is much higher. It can be seen that increasing the input resolution of the classifier no longer reduces the MR_{far} (due to the limited resolution of the analyzed image and the pedestrians appearing on it), but it will significantly slow down the operation of the detection algorithm.

Table 39. MR_{far}, MR_{near} and FPS values achieved for various input resolutions with ACF and CNN detectors for CVC-14 and KAIST datasets

			ACF			CNN		
Dataset	**Set**	**Resolution**	MR_{far} [%]	MR_{near} [%]	FPS*	MR_{far} [%]	MR_{near} [%]	FPS*
CVC-14	1	64×128	30.4	20.1	21.5	26.5	19.7	0.3
	2	56×120	31.6	20.8	22.4	26.3	19.6	0.5
	3	56×112	31.5	20.4	22.4	24.3	17.1	0.5
	4	56×104	31.0	20.9	23.6	23.7	17.6	0.5
	5	48×96	30.3	20.3	25.0	26.5	19.6	0.5
	6	40×88	31.2	19.9	26.0	26.7	19.1	0.8
	7	40×80	31.3	19.8	26.5	25.5	17.4	0.8
	8	40×72	30.6	18.3	26.5	26.8	19.4	1.0
	9	32×64	29.1	17.1	27.1	24.0	17.9	1.3
	10	24×56	29.7	17.4	28.3	26.3	19.7	1.8
	11	24×48	29.8	17.5	28.3	25.6	18.1	2.3
	12	24×40	30.4	19.9	30.4	26.7	19.7	2.8
	13	16×32	34.1	22.8	31.1	28.6	22.4	4.8
KAIST	9	32×64	30.9	28.1	37.3	28.4	23.9	3.5
	10	24×56	29.6	25.9	41.8	30.3	27.7	5.0
	11	24×48	30.1	25.9	56.9	30.9	26.2	5.8
	12	24×40	30.4	25.3	56.9	28.9	23.6	6.5
	13	16×32	30.1	24.3	70.6	33.1	26.3	10.8

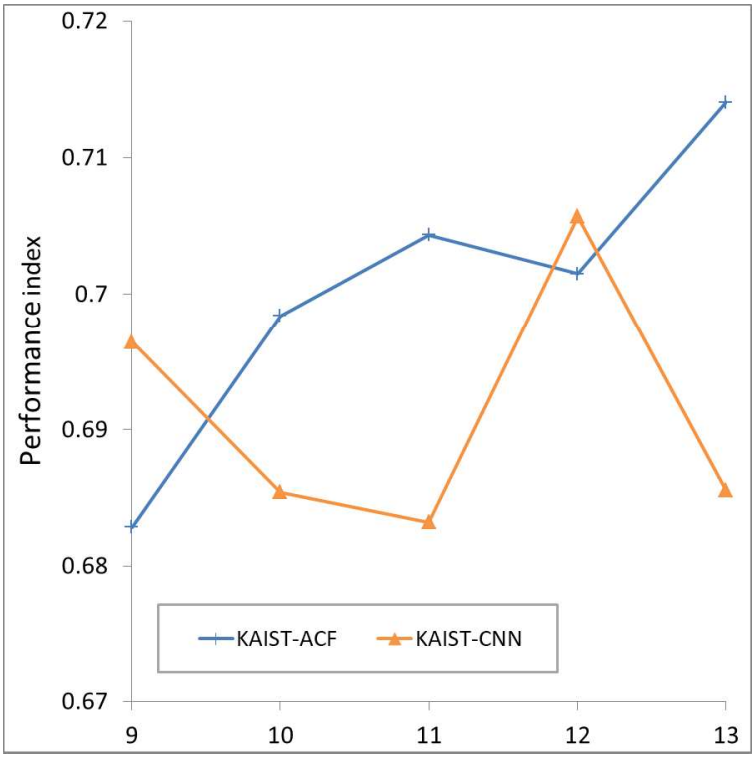
(*) The FPS was calculated for a single-core of Intel Core i7-870 CPU

Table 40. Best performance resolutions for ACF and CNN detectors obtained for CVC-14 and KAIST datasets

Dataset	**Type of Classifier**	**Best Performance Resolution**	**Difference in MR_{far} (*) [%]**	**FPS Acceleration (*) [%]**
CVC-14	ACF	32×64	−1.3	+26.1
	CNN	32×64	−2.5	+333.3
KAIST	ACF	16×32	-0.8	+89.3
	CNN	24×40	+0.5	+85.7

(*) Percentage difference referred to the classifier with the highest resolution.

a)



b)

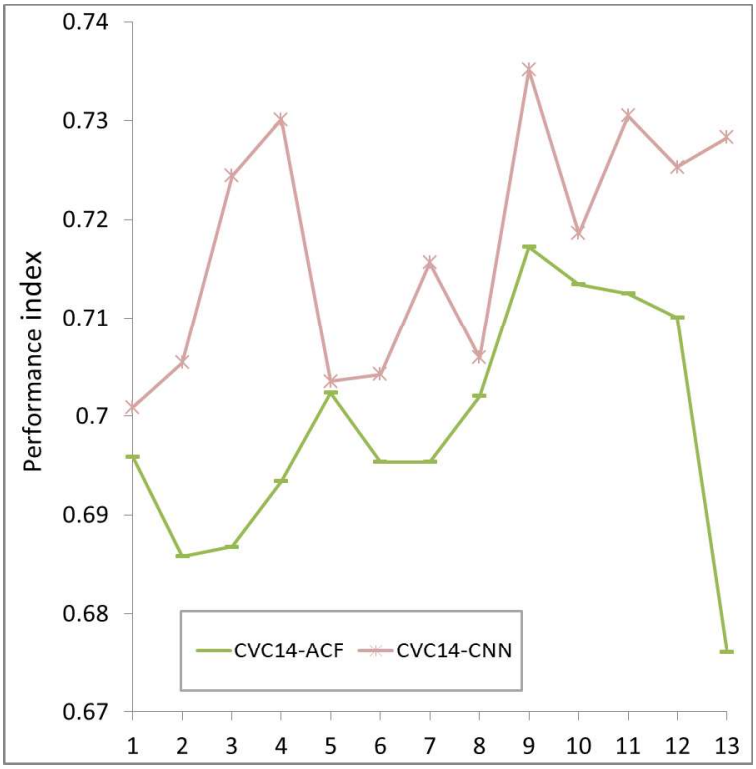


Figure 45. Performance indices as functions of image resolutions (values on the x-axis refer to particular test sets in Table 39) for $w_\rho = 0.95$ and for various detectors and for KAIST (a) and CVC-14 (b) datasets

6.2.4. Adjustment of ROI area

In order to verify the impact of the proposed ROI adjustment method (presented in detail in Chapter 4) on the pedestrian detection accuracy, the experiments were carried out for the scale factor k changing from 1 to 1.4 with the step of 0.05. As a result, 36 measurements of MR_{far} were obtained for both datasets: CVC-14 and KAIST, 9 for each object classifier, namely ACF and deep CNN. Similar to the previous section, experiments for both datasets were performed with double thresholding and with *balanced* settings.

The results are presented in Table 41 and Table 42. Figure 46 and Figure 47. It is noticeable that including the full pedestrian shape in the ROI is important for the object classification stage and for the best-chosen scale of ROI enlargement, MR_{far} may be significantly reduced. In all cases (for both tested detectors and datasets), it was possible to find the minimum value of MR_{far} for scale factor k greater than 1 (see Figure 46 and Figure 47). However, it was achieved for different values of k.

. In case of CVC-14 dataset, the usage of scale factor with k = 1.2 decreased MR_{far} from 29.1% to 24.8% for the ACF classifier and with scale factor k = 1.3 applied for the deep CNN classifier the change of MR_{far} was smaller, but still noticeable, i.e. from 24.0% to 22.4%.

In addition, it can be seen that the CNN-based classifier is less sensitive to the quality of the ROI: It offers more accurate results than the ACF classifier with the same ROIs. The MR_{far} improvement with the scale factor applied is smaller for the CNN. For the ACF detector, any change in scale factor up to 1.4 results in an improvement in MR_{far}. For the CNN-based classifier except the scale factor equal 1.05, all other scales improve the detection accuracy.

In the case of the KAIST dataset, a significant reduction in the value of MR_{far} was obtained for the ACF detector (from 30.1% to 28.1%). However, in the case of the CNN detector, only a slight improvement was obtained (from 28.9% to 28.7%) for the scale factor value k = 1.05. Moreover, for all scale factor k values greater than 1, the ACF detector performed better (achieved lower values of MR_{far}). In the case of the CNN detector, increasing the scale factor k above 1.05 worsened the results (see Table 42 and Figure 47).

In Figure 48 and Figure 49, the modified ROIs with various scale factors are presented. The pedestrian detection results obtained by the ACF classifier are denoted with bounding boxes: red boxes denote no detection, while green boxes denote proper detection. It can be noticed that there is no one optimal scale factor. The results strongly depend on how much the pedestrians were cropped in the initial ROIs. If the mismatch was small, also the small scale factor corrects the erroneous case. If the mismatch was high also the scale factor should be high. Additionally, it could be noticed that for a proper detection, the pedestrian does not have to fit into the ROI in total. However, if the pedestrian in the ROI is too small, it could not be detected (see the case in the third row and the last column in Figure 48).

Additionally, for both classifiers, the impact of the ROI area enlargement on the performance of the entire pedestrian detection process was measured. The detection

time of pedestrians within one image frame increased on average by 0.5 ms with adjustment of segmented ROIs (on a typical PC, using the CPU only). In case of the ACF classifier, the detection time increased by approximately 1.4%, and in case of CNN classifier, the detection time increased by approximately 0.055%.

Finally, it was proved that the proposed solution has a negligible impact on the efficiency of the detection process. This is mainly due to the fact that only the ROI area is increased, not the object classifier input resolution (as presented in Chapter 4, eventually for the classification stage each ROI is resized to one resolution).

Table 41. MR_{far} and MR_{near} for various scale factor k with ACF and CNN detectors for CVC-14 dataset

	ACF		**CNN**	
Scale factor k	MR_{far} [%]	MR_{near} [%]	MR_{far} [%]	MR_{near} [%]
1.4	26.8	19.2	23.2	14.6
1.35	26.5	19.1	23.0	13.2
1.3	26.4	18.2	22.4	13.4
1.25	25.8	17.5	23.0	13.4
1.2	24.8	15.9	22.9	14.7
1.15	25.9	15.6	23.5	15.1
1.1	26.5	15.4	23.5	15.5
1.05	27.3	16.4	24.4	16.4
1.0	29.1	17,1	24.0	17.9

Table 42. MR_{far} and MR_{near} for various scale factor k with ACF and CNN detectors for KAIST dataset

	ACF		**CNN**	
Scale factor k	MR_{far} [%]	MR_{near} [%]	MR_{far} [%]	MR_{near} [%]
1.4	28.8	24.2	35.2	34.7
1.35	28.8	24.3	34.9	34.5
1.3	28.3	23.5	34.4	34.4
1.25	28.1	23.4	32.8	31.2
1.2	28.1	23.2	31.7	29.1
1.15	28.8	23.9	31.1	27.8
1.1	29.4	23.9	29.6	26.1
1.05	30.1	24.5	28.7	21.1
1.0	30.1	24.3	28.9	20.9

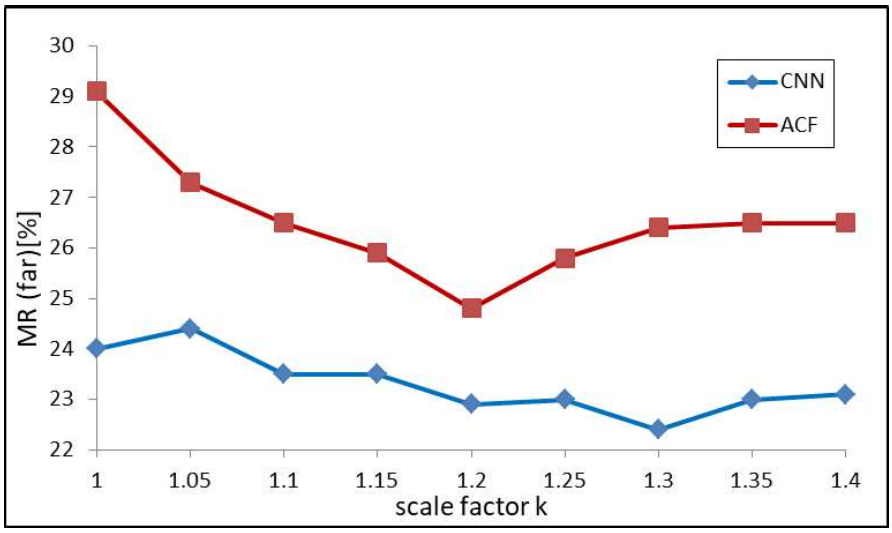


Figure 46. MR_{far} for various values of scale factor k for CVC-14 dataset

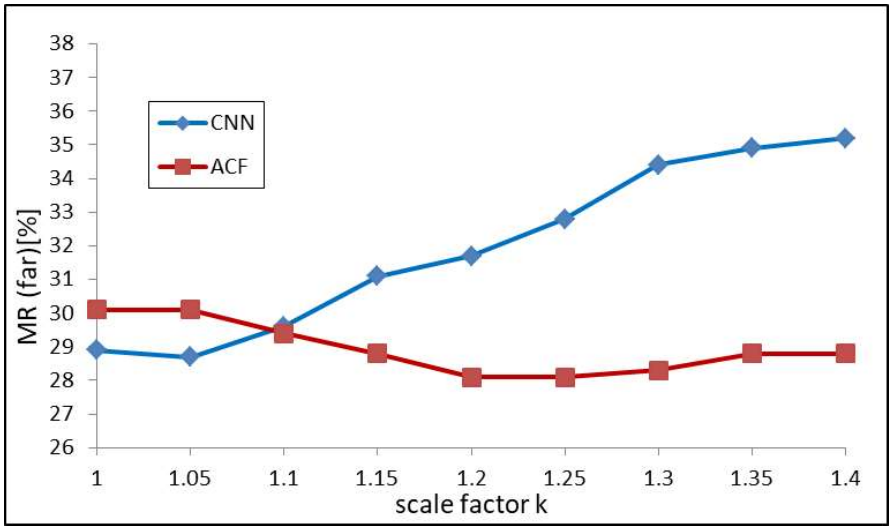


Figure 47. MR_{far} for various values of scale factor k on KAIST dataset

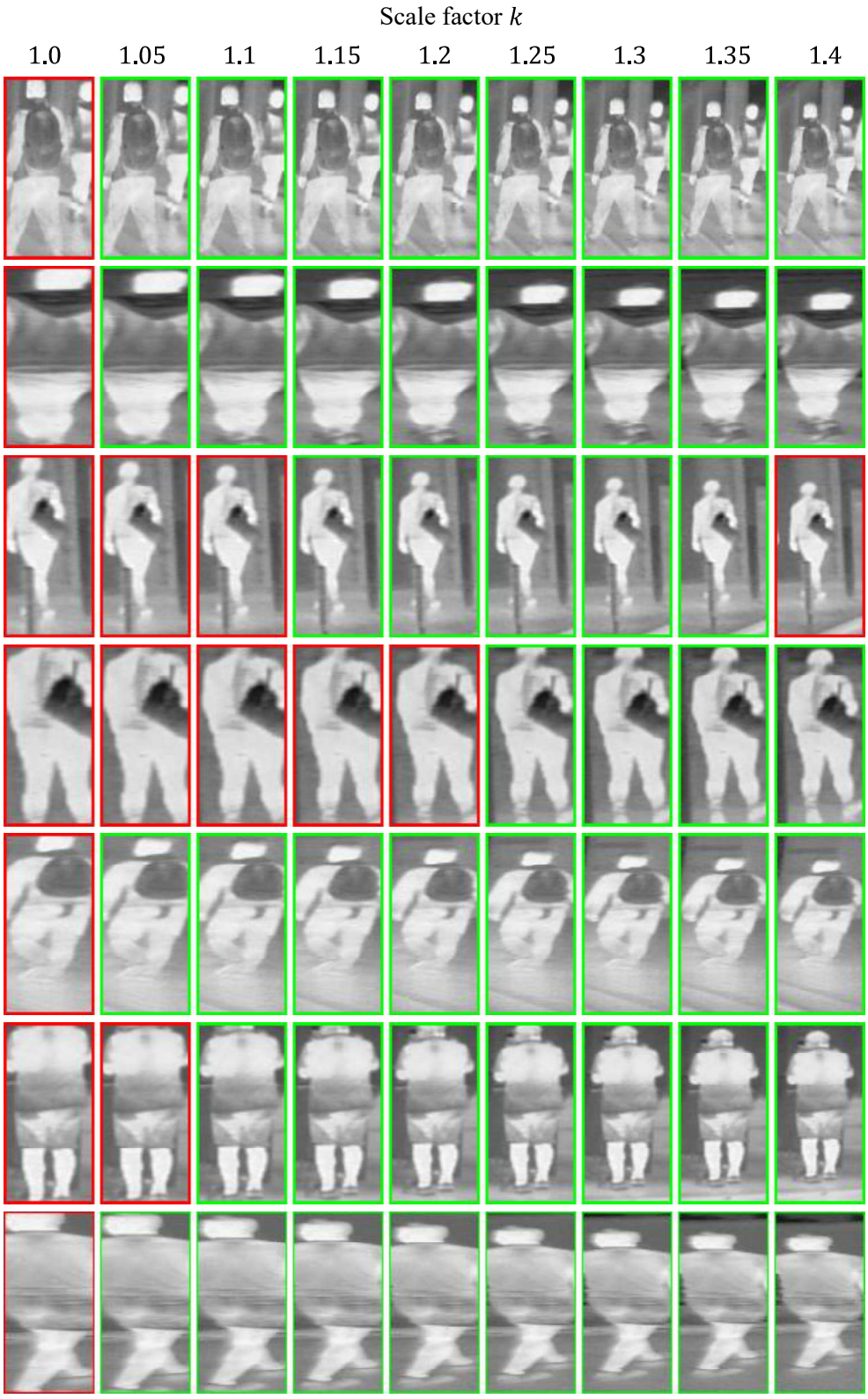


Figure 48. The illustrative examples of ROIs from the CVC-14 dataset for which increasing the ROI area resulted in improved detection result (red bounding box - no detection, green bounding box - correct detection)

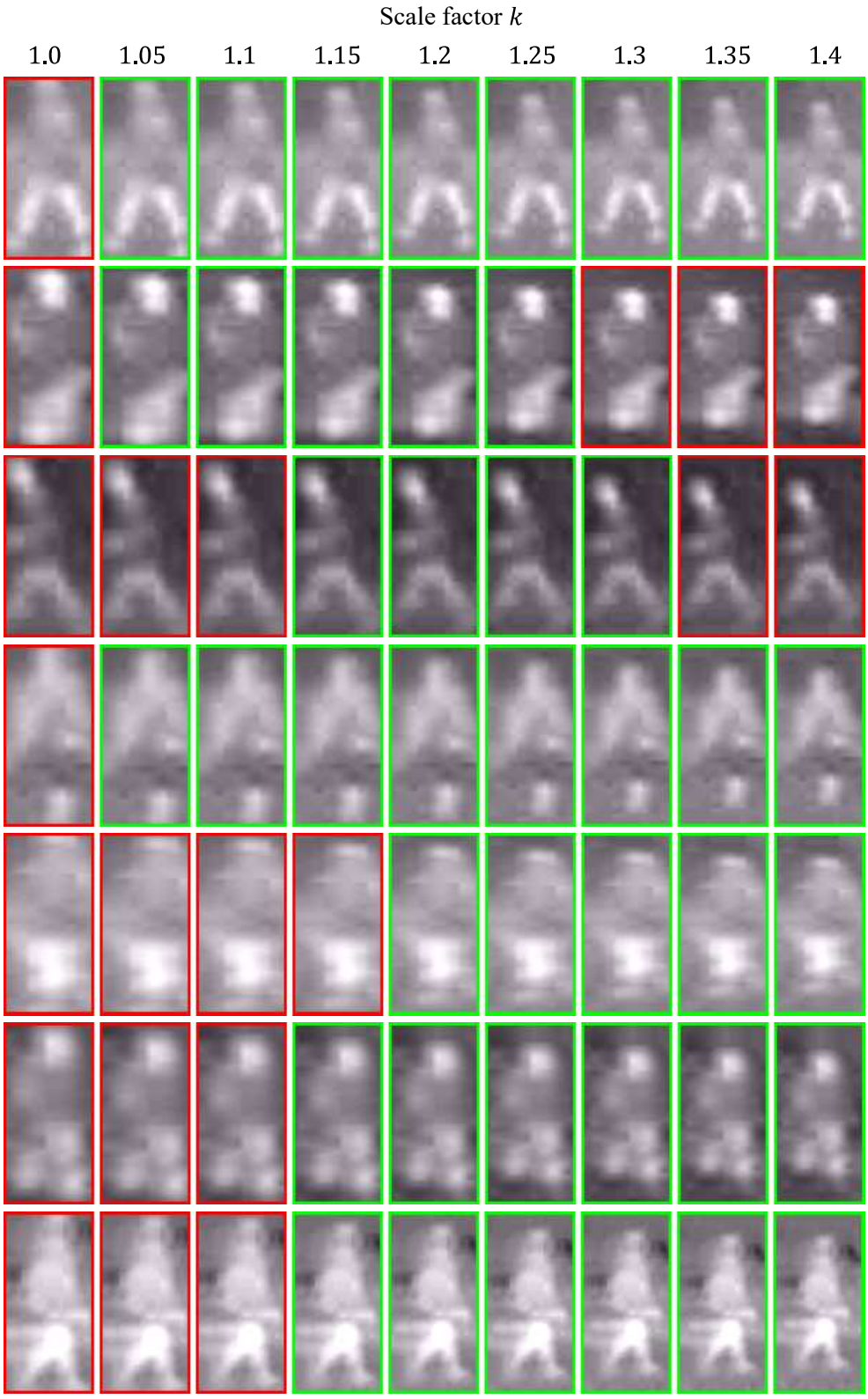


Figure 49. The illustrative examples of ROIs from the KAIST dataset for which increasing the ROI area resulted in improved detection result (red bounding box - no detection, green bounding box - correct detection)

6.2.5. Final results

This section presents extended experiments with the proposed pedestrian detection procedure that were conducted for finally selected settings: selected resolutions and values of scale factor k (obtained in the previous sections), and for all test sequences of KAIST dataset. In addition, the results of MR_{far} and MR_{near} are presented in relation to the FPPI.

The tests were performed for double and triple thresholding for the *balanced* settings of the ROI generation process due to fact that worse results in average were obtained with the *best accuracy* settings in the initial experiments (presented in Table 38). The results of the experiments are presented in Table 43 (for the CVC-14 dataset) and in Table 44 (for the KAIST dataset). In addition, Figures 50-53 show graphs presenting the achieved MR_{far} and MR_{near} values in relation to the FPPI for both datasets, and Figures 54 and 55 show illustrative examples of detection results obtained with the proposed pedestrian detection approach.

Table 43. Final values of MR_{far}, MR_{near} and FPS obtained for the CVC-14 dataset with ACF and CNN detectors

Dataset	Parameter	Double thresholding		Triple thresholding	
		ACF	CNN	ACF	CNN
CVC-14	MR_{far} [%]	24.8	22.4	28.9	28.2
	MR_{near} [%]	15.9	13.4	17.3	15.9
	FPS*	27.0	1.3	17.7	0.8
	FPS**	91.0	4.5	60.4	2.3

(*) The FPS was calculated for single-core of Intel Core i7-870 CPU
(*) The FPS was calculated for four-core of Intel Core i7-870 CPU

In the case of the CVC-14 dataset, the obtained results of MR_{far} and MR_{near} improved compared to the results presented in Table 38. Detection with the double thresholding method still achieves better results than detection with the tripe thresholding (e.g., for double thresholding with an ACF detector, the value of MR_{far} = 24.8%, and for the triple thresholding, the value of MR_{far} = 28.9%).

The graphs of the values of MR_{far} and MR_{near} in relation to the FPPI presented in Figures 50 and 51 show that for lower FPPI values the ACF detector achieves better results than the CNN detector (despite worse values achieved for FPPI = 1). Moreover, for the ACF detector, both values of MR_{near} and MR_{far} decreases evenly with increased FPPI value. The situation is different in the case of the CNN detector: for very small FPPI values, the MR_{far} is lower than MR_{near}.

For the KAIST dataset (set 09 - campus), the final obtained results are much better than the initial values (presented in table 38). For all settings, the values of MR_{far} and MR_{near} decrease (e.g., for double thresholding with ACF detector MR_{far} decreases from 30.9% to 28.1%, and for triple thresholding with ACF detector MR_{near} decreases from 20.1% to 14).

The results presented in Table 44 show that for all subsets of the KAIST dataset, the detection with the triple thresholding reaches significantly lower values of MR_{far} than for the double thresholding.

It can also be seen that significantly better MR_{far} values are achieved for the subset 09 (campus) and the subset 10 (roadway) than for the subset 11 (downtown). This is because the detection for the subset 11 is more difficult due to a greater traffic and a lower thermal contrast. The averaged results for the entire KAIST dataset (weighted average depending on the number of image frames) are also presented in the bottom of Table 44.

In most cases, the ACF detector achieves better values than the CNN detector. The graphs of MR_{far} and MR_{near} in relation to the FPPI (presented in Figures 52 and 53) have similar waveforms for both detectors. However, for the CNN detector for low FPPI values, the value of MR_{near} is higher than the MR_{far}.

For both datasets, the obtained values of FPS for the proposed pedestrian detection algorithm are very high using CPU processing only (up to 91 FPS with ACF detector for CVC-14 dataset and up to 261.2 FPS for the KAIST dataset). For the CNN detector, the obtained FPS values are much lower than for ACF (due to the complex structure of the detector), but still close to the real-time processing.

Table 44. Final values of MR_{far}, MR_{near} and FPS obtained for the KAIST dataset (all test sequences) with ACF and CNN detectors

Set no.	**Parameter**	**Double thresholding**		**Triple thresholding**	
		ACF	CNN	ACF	CNN
Set 09 (campus)	MR_{far} [%]	28.1	28.1	27.2	26.1
	MR_{near} [%]	21.1	21.1	14.6	16.2
	FPS*	70.6	6.5	59.4	5.2
	FPS**	261.2	17.6	219.8	14.4
Set 10 (roadway)	MR_{far} [%]	32.5	33.9	29.2	30.7
	MR_{near} [%]	18.6	22.0	16.7	20.8
	FPS*	37.0	3.5	32.7	2.5
	FPS**	136.9	9.7	121.2	7.0
Set 11 (downtown)	MR_{far} [%]	47.6	49.3	44.3	46.2
	MR_{near} [%]	37.2	38.6	27.7	32.8
	FPS*	18.1	1.7	16.3	1.3
	FPS**	67.0	4.8	60.3	3.6
Average	MR_{far} [%]	34.9	36.2	32.2	33.1
	MR_{near} [%]	23.3	25.5	18.7	22.5
	FPS*	40.2	3.8	34.9	2.8
	FPS**	148.6	10.3	129.2	7.9

(*) The FPS was calculated for single-core of Intel Core i7-870 CPU

(*) The FPS was calculated for four-core of Intel Core i7-870 CPU

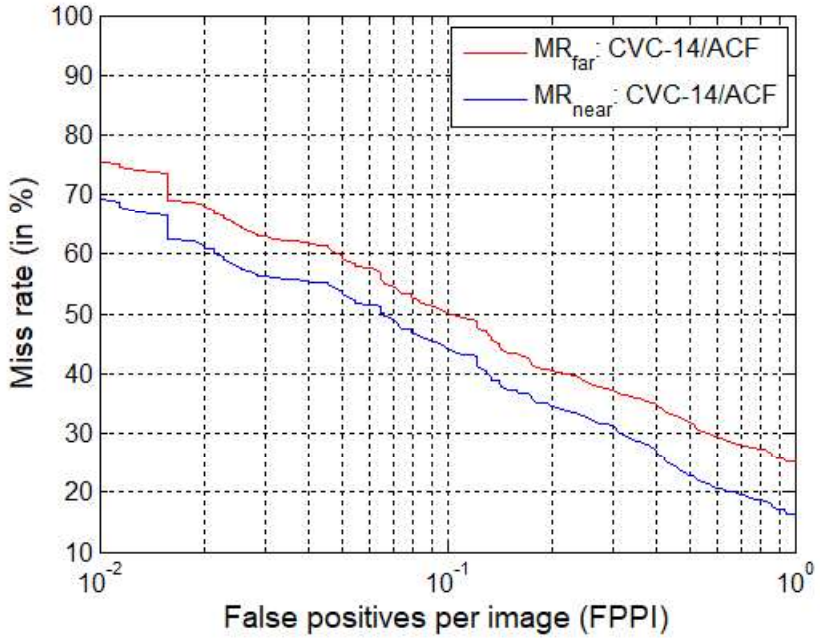


Figure 50. MR_{far} and MR_{near} in relation to the FPPI parameter obtained with the optimal settings for CVC-14 dataset with the ACF detector

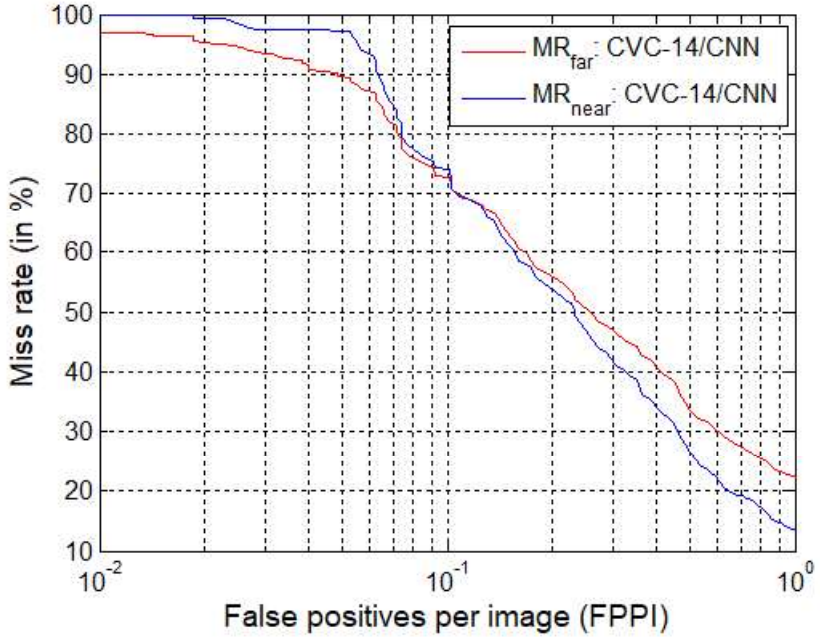


Figure 51. MR_{far} and MR_{near} in relation to the FPPI parameter obtained with the optimal settings for CVC-14 dataset with the CNN detector

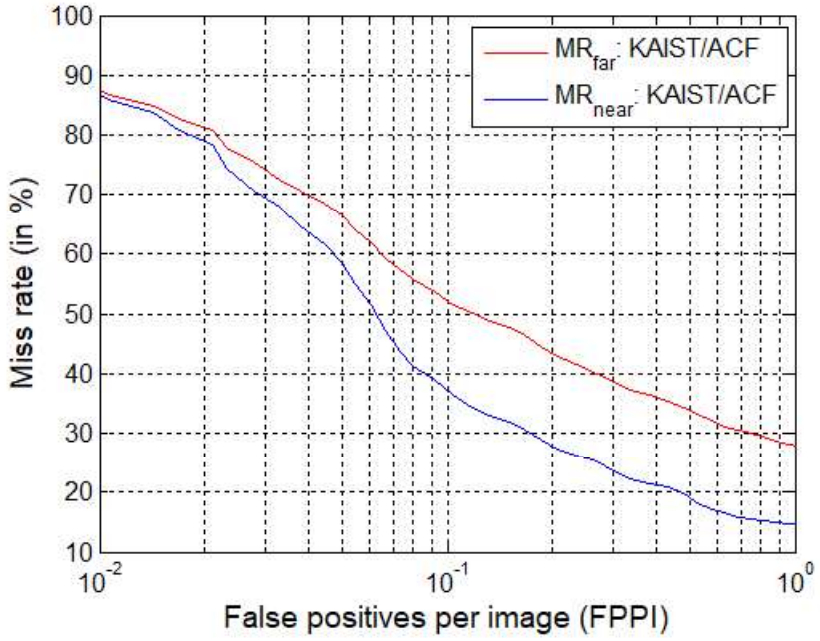


Figure 52. MR_{far} and MR_{near} in relation to the FPPI parameter obtained with the optimal settings for the KAIST dataset (set 09) with the ACF detector

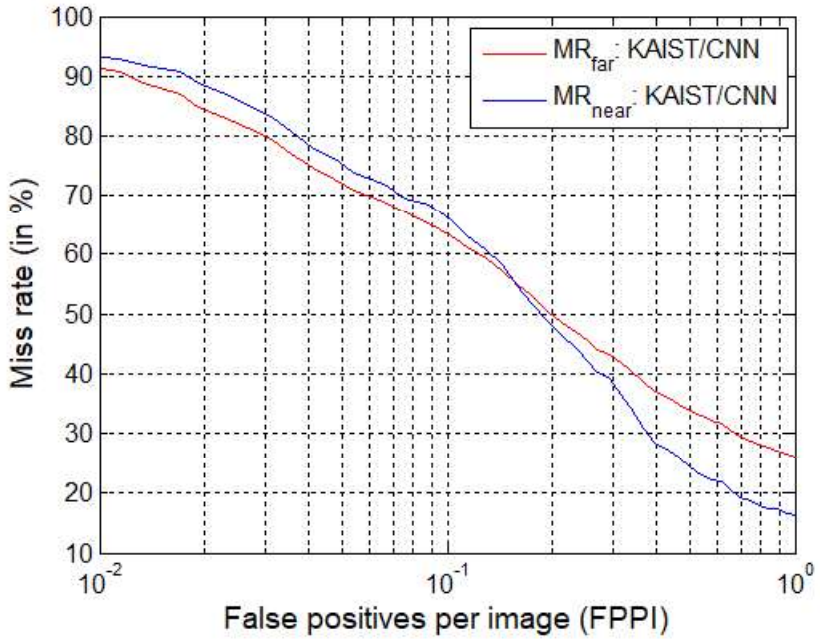


Figure 53. MR_{far} and MR_{near} in relation to the FPPI parameter obtained with the optimal settings for the KAIST dataset (set 09) with the CNN detector

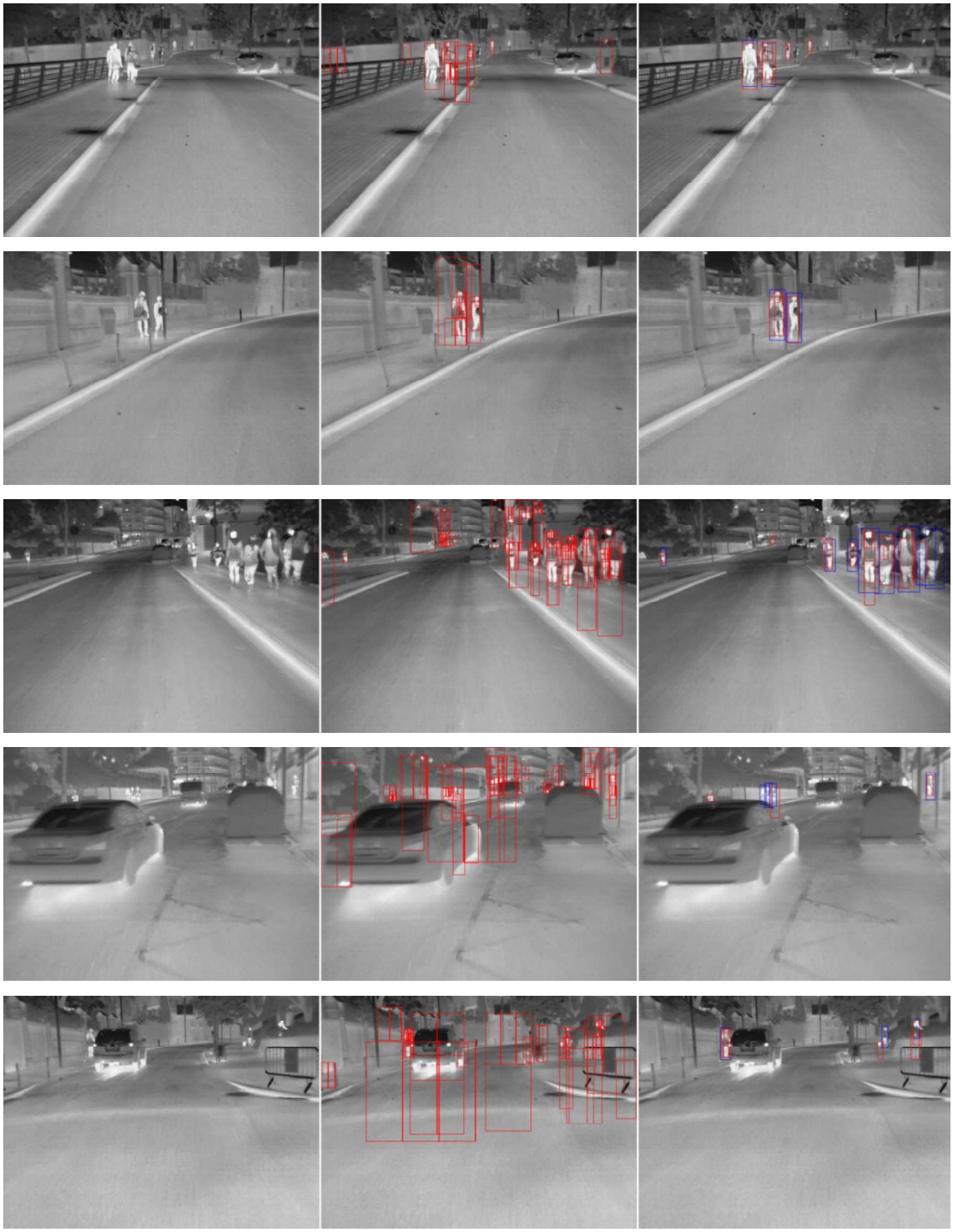


Figure 54. The illustrative examples of pedestrian detection obtained with ACF detector for the CVC-14 dataset, from left: input thermal image, thermal image with marked ROIs (red boxes), thermal image with marked detections (red boxes) and annotations (blue boxes)

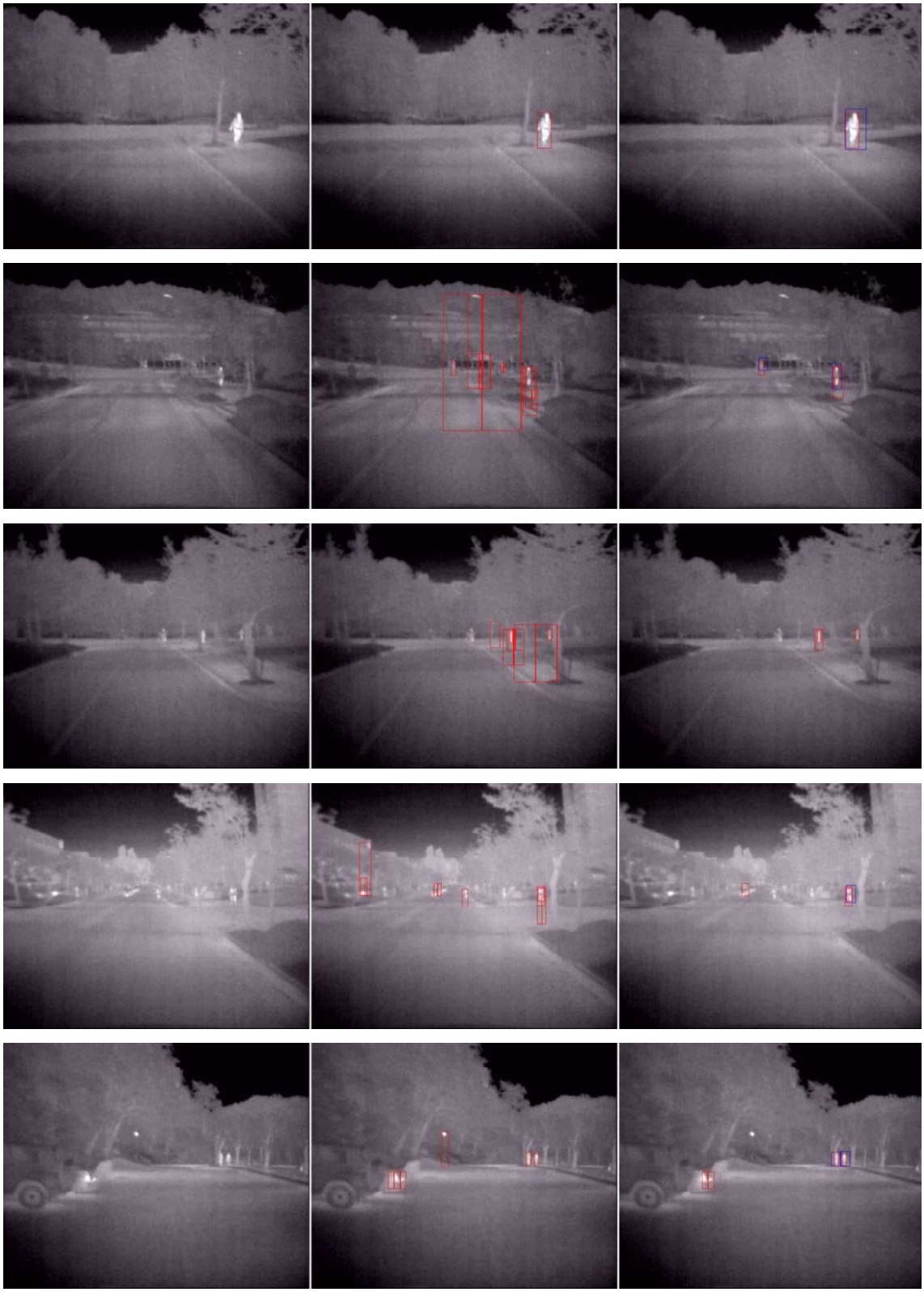


Figure 55. The illustrative examples of pedestrian detection obtained with ACF detector for the KAIST dataset (set 09), from left: input thermal image, thermal image with marked ROIs (red boxes), thermal image with marked detections (red boxes) and annotations (blue boxes)

6.3. Comparison of results

This section compares the results obtained in the previous section with the other techniques found in the literature for the CVC-14 and KAIST datasets. The primary objective of the comparison was to verify whether the pedestrian detection algorithm with the proposed improvements achieves better results than similar solutions (based on the same detectors) but without the introduced improvements.

In the beginning, comparative tests were carried out on the pedestrian detection process with the proposed ROI generation technique and with the sliding window technique. The experiments were conducted for the same detectors: ACF and CNN, but with different ROI generation techniques. The results of the comparison are presented in Table 45.

Due to the same implementation, parameters values of tested detectors, and the same methodology of evaluating the results, it was possible to objectively evaluate the accuracy and performance of the detectors with and without the proposed improvements.

The results presented in Table 45 show that for both the CVC-14 dataset and the KAIST datasets, the values of the parameters MR_{far} and MR_{near} are much lower for pedestrian detection based on the proposed ROI generation technique (e.g. MR_{far} = 22.4% for the CVC-14 dataset and MR_{far} = 32.2% for the KAIST dataset) than for the sliding window technique (MR_{far} = 44.7% for the CVC-14 dataset and MR_{far} = 45.8% for the KAIST dataset).

Although in general, the sliding window technique has low MR value at the ROI generation stage (close to 0, for the assessment of the ROI generation stage itself, as presented in Chapter 3), a very large number of ROIs obtained with this technique (on average for one image frame it is 6132 ROIs for the CVC-14 dataset and 6321 for KAIST dataset) causes that the classifier makes more false detections (false-positives). As a result, in order to obtain the desired FPPI value, the detection threshold must be heightened, which causes a significant increase in the values of MR_{far} and MR_{near}.

Table 45. Comparison of pedestrian detection results based on the proposed ROI generation technique and the sliding window approach for CVC-14 and KAIST datasets

Dataset	ROI generation	Detector	$\mathbf{MR_{far}}$ **[%]**	$\mathbf{MR_{near}}$ **[%]**	$\mathbf{LAMR_{far}}$ [%]	$\mathbf{LAMR_{near}}$ [%]	**FPS**
CVC-14	Sliding window	ACF	44.7	38.9	68.9	57.3	2.7
	Proposed	ACF	24.8	15.9	**49.7**	**42.2**	**91.0**
	Sliding window	CNN	45.1	36.7	73.1	72.8	0.2
	Proposed	CNN	**22.4**	**13.4**	64.1	64.5	4.5
KAIST	Sliding window	ACF	45.8	42.4	72.4	66.2	4.1
	Proposed	ACF	**32.2**	**18.7**	**58.7**	**48.7**	**129.2**
	Sliding window	CNN	45.9	44.3	74.2	74.6	0.4
	Proposed	CNN	33.1	22.5	61.6	62.2	7.9

In addition, the computational efficiency of the pedestrian detection is much higher (even faster by 125 FPS) based on the proposed ROI generation technique and reaches a value up to 130 FPS. For this reason, it can be concluded that the operation of the proposed pedestrian detection algorithm in the real-time is possible even in a vehicle or an embedded system,

In the next step, pedestrian detection results were compared with similar solutions presented in the literature. As mentioned earlier, the purpose of the comparison was to verify whether the pedestrian detection algorithm with the proposed improvements achieves better results than the detection algorithm without the introduced improvements. Therefore only similar solutions were considered for the comparison (based on the same or similar detectors, i.e., ACF and AlexNet/CNN).

In order to facilitate the comparison and to separate the parameters $\mathrm{MR}_{\mathrm{far}}$ and $\mathrm{MR}_{\mathrm{near}}$ form the value of the FPPI parameter an additional parameter, called log-average miss rate (LAMR), was used. This parameter is an average of the measured miss rate in the range of 10^{-2} to 10^{0} FPPI and is often used in the literature to compare the overall detector performance. Its value was calculated for the proposed algorithm and presented in Table 45. Finally, Table 46 and Table 47 present a comparison of results based on the LAMR and FPS parameters for CVC-14 and KAIST datasets.

For the comparison presented in Table 46 and Table 47, the results of similar implementations based on ACF detectors were selected. The comparison of the tested CNN / AlexNet detector was not made due to the lack of results for the CNN detectors based on this architecture for the KAIST and CVC-14 datasets. For thermal pedestrian detection based on CNN in the literature, the results are mainly presented for the VGG16 and ResNet-50 models [99], [100].

In the case of the KAIST dataset, the results presented in the literature refer to a *reasonable* test set proposed in [26]. For this set only pedestrians with a height greater than 55 pixels are taken into account. For this reason, the results from the literature (presented in Table 46) should be compared with the values of the parameter $\mathrm{LAMR}_{\mathrm{near}}$ (for which the analysed pedestrians have height greater than 40 pixels) of proposed pedestrian detection procedure than with the value of $\mathrm{LAMR}_{\mathrm{far}}$ (for which the analysed pedestrians have height greater than 20 pixels).

The results presented in the literature for the closest version of the tested ACF detector implemented in this study: "ACF + T" [23] are worse (LAMR = 74.5%) than those obtained for the pedestrian detector with the proposed improvements ($\mathrm{LAMR}_{\mathrm{near}}$= 48.7%). The LAMR value for the "ACF + T" detector is close to the values presented in this work for the tested detector based on the sliding window technique (LAMR = 72.4%, cf. results presented in Table 45).

The remaining compared implementations achieve lower LAMR values (cf. Table 46) but have different improvements in the object classification step. These include multispectral pedestrian detection (mainly RGB + thermal), which significantly improves the accuracy of pedestrian detection [99], [100]. However, also in this case the

achieved LAMR value of ACF detector with proposed ROI generation technique is lower or comparable to these multispectral solutions (cf. Table 46).

Table 46. Comparison of results with similar detectors for KAIST dataset (night-time test set)

Reference	Pedestrian detector	**LAMR** [%] ($\mathrm{LAMR_{far}}$ / $\mathrm{LAMR_{near}}$)	FPS
this work	Proposed ROI / ACF	(58.7 / **48.7**)	129.2 (CPU)
this work	Proposed ROI / CNN	(60.6 / **61.2**)	7.9 (CPU)
[26]	ACF+T	74.5	N.A.
[26]	ACF+T+TM+TO	64.9	N.A.
[26]	ACF+T+THOG	63.9	N.A.
[101]	ACF-RGBT+THOG	61.5	N.A.
[102]	ACF	56.2	0.4 (CPU)
[39]	Multispectral ACF	48.2	N.A.

(*) N.A. – not available

Table 47. Comparison of results with similar detectors for CVC-14 dataset (night-time test set)

Reference	Pedestrian detector	**LAMR** [%] ($\mathrm{LAMR_{far}}$ / $\mathrm{LAMR_{near}}$)	FPS
this work	Proposed ROI / ACF	(**49.7** / 42.2)	91.0 (CPU)
this work	Proposed ROI / CNN	(**64.1** / 64.5)	4.5 (CPU)
[39]	Multispectral ACF	65.4	N.A.
[103]	Multispectral ACF (reimplementation)	48.2	N.A.

(*) N.A. – not available

The computational efficiency of the pedestrian detection with the proposed ROI generation technique is much higher than presented in [102] for the ACF detector (the only cited paper with information about the computational efficiency), which is only 0.4 FPS. This value is also lower than the tested implementation of the detector based on the sliding window technique (4.1 FPS, cf. Table 45), which is probably due to the multispectral detection approach and the higher input resolution of the ACF detector.

In the case of the CVC-14 dataset, the comparison of the achieved LAMR values for both detectors was possible only with solutions based on multispectral imaging (RGB + thermal). As mentioned before, these solutions achieve lower LAMR values than solutions based on a single input source. However, the achieved results for proposed detector are better (LAMR = 49.7% for the ACF detector) than for the Multispectral ACF [39] (LAMR = 65.4%) or close to results presented in [103] (LAMR = 48.2%). For the compared detectors, it was not possible to obtain information about computational efficiency from the papers [39], [103].

Concluding, the obtained detection accuracy for the ACF detector is comparable to multispectral detectors (see Table 46 and Table 47). Further reduction of the LAMR value could be possible by improving object detection stage by: additional data augmentation, multispectral classification, additional tracking step, using better detectors, e.g., Checkerboards [77] or more complex CNN models such as VGG16 or ResNet-50 [99], [100]. However, the main aim in this chapter was to verify the usefulness of the proposed improvements, and that was possible with the tested ACF and AlexNet / CNN detectors (for the reasons described at the beginning of this chapter).

7. Multi-spectral imaging for CCTV operators

In this chapter, an additional option of multi-spectral imaging for CCTV operators is presented. At the beginning of this chapter, a method of creating multi-spectral images is described. Then the experiments with groups of observers are presented, performed to test the efficiency of the proposed multi-spectral imaging.

7.1. Multi-spectral imaging

According to observations following from Figure 56, to facilitate analysis of CCTV images at night by humans (for example, by CCTV operators), it is proposed to use the multi-spectral image quality, which is obtained by merging the conventional camera image with its thermal camera image counterpart [104]. Both cameras should operate in parallel and observe the same scene. A similar idea was already proposed by Flir company as the so-called "multi-spectral dynamic imaging function" offered in the measurement cameras [105].

Due to a low number of other but important details in IR images (Figure 56b), for manual scene analysis by e.g. the CCTV operators, more convenient are the multi-spectral images (Figure 56c), which are obtained by merging conventional images with their IR counterparts.

Indeed such important details, e.g. road lines, posts and signs, lights of upcoming cars, etc., are altogether much better visible by humans in Figure 56c than in Figure 56a or Figure 56b, separately.

a) b) c)

Figure 56. Illustrative images of the same scene recorded at night: (a) visual light image, (b) thermal image, (c) multi-spectral image.

The proposed multi-spectral option for the CCTV visualization (see Figure 57) is realized as follows: first, the thermal image is upsized to the resolution of the conventional image (as typically IR cameras offer lower resolutions as the visible light cameras), then the conventional image is taken as a background for the final multi-

spectral image. After that, all high luminance pixels in the thermal image (those exceeding a certain threshold) replace the corresponding background pixels, but only if their luminances are higher than in the conventional image.

7.2. Experiments

To study the effectiveness of the multi-spectral perception by humans, experiments with a group of observers were performed. The observers had to count pedestrians in images of three types: conventional, thermal, and multi-spectral. Additionally, they should estimate the number of pedestrians located on the roadway. The first task was proposed to evaluate the precision and speed of pedestrian detection using the analysed image types, while the second task aimed to check the ability of correct environmental location of the detected pedestrians.

A testing set consisted of 11 different monitoring scenes of resolutions 640×320. For each scene, images of three types were prepared: conventional, thermal, and multi-spectral. Images of the first two types were taken directly from the USArmy Tetravision dataset [37]. Images of the third type were generated using the approach described above. Finally, the testing set was composed of 33 images.

The experiment participants were divided into two groups. The first group consisted of 45 untrained persons (students), and the second group was constituted by 14 trained observers (most of them from the academic staff). The experiment was performed with specially prepared software and in similar lighting conditions.

With three scenes (rows in Figure 57), visibility enhancement using the multi-spectral option is illustrated. The first example shows a single pedestrian at a short distance from the camera. The second one presents pedestrians in a far distance (both are invisible with the standard camera). The last example depicts two pedestrians in the mid-range. Only one of them (the right one) is clearly visible in the standard camera image.

The results of the performed experiments are presented in Table 48. It can be seen that for precise counting of pedestrians, thermal images are much better than conventional images (an improvement from 55–60% to ca. 98%) and even better than multi-spectral images. However, the precise localization of pedestrians (in this case of those present in the road) is the best using the proposed multi-spectral image quality (an improvement from ca. 53% to ca. 87%).

The performed experiments with observers (presented in Table 48) show that the proposed option of multi-spectral imaging (obtained by merging conventional and thermal camera images) effectively improves the manual CCTV scene analysis at night, shortens reactions and supports faster identification of objects.

Table 48. Influence of type of night-vision imaging on precision and speed of analysis of monitoring situations

Analysis	Description of the result	Conventional images		Thermal images		Multi-spectral images	
		1	2	1	2	1	2
General pedestrian counting	correct answers	55.2%	58.2%	98.1%	98.3%	89.9%	91.2%
	mean time of counting	2.6 s	2.0 s	2.1 s	1.7 s	2.4 s	2.1 s
Pedestrian counting on the roadway	correct answers	52.3%	53.0%	68.7%	71.6%	80.5%	87.3%

1 – Results obtained with untrained observers.
2 – Results obtained with trained observers.

a) b) c)

Figure 57. Illustrative examples of three scenes (in each row) from the prepared testing set: (a) conventional camera image, (b) thermal image, (c) prepared multi-spectral image (yellow rectangle denotes the field of view of the thermal camera).

8. Conclusions

In this dissertation, issues concerning night-vision pedestrian detection were considered. The author proposed effective solutions in three parts: first, a new ROI generation approach for the thermal images based on image thresholding, second, the technique of additional ROI adjustment (slightly enlarging the ROI area of the image) before the object classification stage, third, the procedure for tuning of the object classification process with the universal performance index. These solutions were designed to improve the accuracy of the pedestrian detection process and to achieve real-time performance in order to apply it in vehicles (such as ADAS equipped cars or autonomous vehicles).

At the beginning, it was pointed out that it is potentially possible to use segmentation by thresholding on thermal images at night. For this reason, the new ROI generation method was proposed. This method performs image segmentation multiple times with different threshold values, then the set of ROIs is extended with new additional areas with the regions enlargement technique and finally filtered with the proposed set of candidates selection techniques.

The results obtained for both public datasets: CVC-14 and KAIST allow to conclude that it is possible to accurately and efficiently perform the segmentation of thermal images at night through the thresholding. Very low MR values were achieved: 1.2% for the CVC-14 dataset and 8.8% for the KAIST dataset, still offering very high computational efficiency (varying from 44 to 347 FPS depending on the settings) obtained using only the CPU.

Next, the technique of additional ROI adjustment was proposed to address the problem of inaccurate ROI adjustment. Inaccurate matching the edges of ROI to the outer edges of the pedestrian may lead to cases of not a whole pedestrian covered with the ROI. Such too small ROIs may finally be rejected by the classifier. The results presented for this technique show that it was possible to value for tested datasets. For example, in the CVC-14 dataset, with the ACF classifier MR_{far} decreased from 29.1% to 24.8%. The proposed solution has a negligible impact on the computational time of detection process mainly due to the fact that only the ROI area is increased, not the object classifier input resolution.

In the third part of this dissertation, tuning of the object classification stage was considered. It was pointed out that the classifiers are often used without an adaptation of their input resolution to the resolution of the specific dataset or camera, especially in the solutions with a complicated structure like deep convolutional neural networks.

The specialized procedure for tuning of the object classification stage was proposed. This procedure is based on a novel and universal performance index. Using this procedure, the author demonstrates that properly tuning of the object detection stage to the analysed image source, e.g., to the sensor type, camera perspective and the resolution of image is important and significantly affects the computational performance. The results of experiments show that the properly tuned detectors achieve good detection accuracy even for relatively low resolutions. It can be seen that increasing the input resolution of the classifier above a certain level no longer increases

the detection accuracy, but will significantly slow down the operation of detection algorithm. Generally, the presented approach can be applied not only to the considered problem but it can be adapted to detection of any type of object with any classifier.

In the fourth part of this dissertation, the whole pedestrian detection algorithm based on the proposed improvements was tested. These tests were carried out for two object detectors, namely ACF and AlexNet/CNN. The comparison of the results was performed for the pedestrian detection based on the proposed ROI generation technique with detection based on the sliding window technique and with the results presented in the literature.

It was proved that the pedestrian detection based on proposed ROI generation approach (based on thresholding of thermal images) is more accurate than detection based on the sliding window technique and achieves much higher computational performance (even several dozen times faster, with up to 130 FPS for using CPU only). Furthermore, the comparison with similar object classification methods presented in the literature shows that the proposed approach achieves better results (i.e. lower LAMR value and much higher FPS). The obtained detection accuracy for the single ACF detector is comparable to much complicated multispectral detectors.

The multi-spectral imaging as an option for CCTV operators was the last part of this dissertation. The multi-spectral images were obtained by merging the conventional camera image with its thermal camera image counterpart. The experiments performed with observers show that the multi-spectral imaging effectively improves the manual CCTV scene analysis at night, shortens reactions and supports faster identification of objects.

Concluding, the author's proposed improvements of the night-vision pedestrian detection procedure increase the detection accuracy and computational efficiency. Therefore, the scientific aim of this Ph.D. dissertation has been accomplished and the scientific thesis, namely: "The developed approach of night-vision pedestrian detection based on proposed ROI generation by thresholding of thermal images and by properly tuned object classification procedure improves detection accuracy and significantly increases computational efficiency of the pedestrian detection process" has been proven.

The proposed pedestrian detection system can be applied in various vehicles, driver assistance systems, and autonomous cars. In case of safety-critical applications, it is recommended to support the proposed system by some other detection systems operating with different sensors in order to increase the final reliability of the system.

References

[1] “Annual Accident Report 2018,” European Commission. [Online]. Available: https://ec.europa.eu/transport/road_safety/road-safety-facts-figures-0_en

[2] J. F. Pace, J. Sanmartin, P. Thomas,, *et al.*, “Basic Fact Sheet Pedestrians,” Deliverable D3.9 of the EC FP7 project DaCoTA, 2012. [Online]. Available: https://ec.europa.eu/transport/road_safety/sites/roadsafety/files/pdf/statistics/dacota/bfs2012_dacota-intras-pedestrians.pdf

[3] S. Plainis, “Road traffic casualties: understanding the night-time death toll,” *Inj. Prev.*, vol. 12, no. 2, pp. 125–138, Apr. 2006, doi: 10.1136/ip.2005.011056.

[4] “POLICY ORIENTATIONS ON ROAD SAFETY 2011-2020,” European Commission, Jul. 2010. [Online]. Available: http://erscharter.eu/resources-knowledge/open-library/policy-orientations-road-safety-2011-2020_en

[5] K. Schreiner, “Night Vision: infrared takes to the road,” *IEEE Comput. Graph. Appl.*, vol. 19, no. 5, pp. 6–10, Oct. 1999, doi: 10.1109/38.788791.

[6] “Honda Develops World’s First Intelligent Night Vision System Able to Detect Pedestrians and Provide Driver Cautions-Available on Legend model to be released in Fall 2004.” [Online]. Available: https://global.honda/newsroom/news/2004/4040824a-eng.html

[7] Q. Liu, J. Zhuang, and S. Kong, “Detection of pedestrians at night time using learning-based method and head validation,” in *2012 IEEE International Conference on Imaging Systems and Techniques Proceedings*, Jul. 2012, pp. 398–402. doi: 10.1109/IST.2012.6295596.

[8] A. Hosseini, D. Bacara, and M. Lienkamp, “A system design for automotive augmented reality using stereo night vision,” in *2014 IEEE Intelligent Vehicles Symposium Proceedings*, MI, USA, Jun. 2014, pp. 127–133. doi: 10.1109/IVS.2014.6856484.

[9] S. J. Krotosky and M. M. Trivedi, “On Color-, Infrared-, and Multimodal-Stereo Approaches to Pedestrian Detection,” *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 4, pp. 619–629, Dec. 2007, doi: 10.1109/TITS.2007.908722.

[10] K. Makantasis, A. Nikitakis, A. D. Doulamis, N. D. Doulamis, and I. Papaefstathiou, “Data-Driven Background Subtraction Algorithm for In-Camera Acceleration in Thermal Imagery,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 9, pp. 2090–2104, Sep. 2018, doi: 10.1109/TCSVT.2017.2711259.

[11] M. Bertozzi, A. Broggi, M. Del Rose, M. Felisa, A. Rakotomamonjy, and F. Suard, “A Pedestrian Detector Using Histograms of Oriented Gradients and a Support Vector Machine Classifier,” in *2007 IEEE Intelligent Transportation Systems Conference*, Sep. 2007, pp. 143–148. doi: 10.1109/ITSC.2007.4357692.

[12] G. Li, S. E. Li, R. Zou, Y. Liao, and B. Cheng, “Detection of road traffic participants using cost-effective arrayed ultrasonic sensors in low-speed traffic situations,” *Mech. Syst. Signal Process.*, vol. 132, pp. 535–545, Oct. 2019, doi: 10.1016/j.ymssp.2019.07.009.

[13] H. Rohling, S. Heuel, and H. Ritter, “Pedestrian detection procedure integrated into an 24 GHz automotive radar,” in *2010 IEEE Radar Conference*, May 2010, pp. 1229–1232. doi: 10.1109/RADAR.2010.5494432.

[14] T. Ogawa, H. Sakai, Y. Suzuki, K. Takagi, and K. Morikawa, “Pedestrian detection and tracking using in-vehicle lidar for automotive application,” in *2011 IEEE Intelligent Vehicles Symposium (IV)*, Jun. 2011, pp. 734–739. doi: 10.1109/IVS.2011.5940555.

[15] J. Schlosser, C. K. Chow, and Z. Kira, “Fusing LIDAR and images for pedestrian detection using convolutional neural networks,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 2198–2205. doi: 10.1109/ICRA.2016.7487370.
[16] C. Premebida, O. Ludwig, and U. Nunes, “Exploiting LIDAR-based features on pedestrian detection in urban scenarios,” in *2009 12th International IEEE Conference on Intelligent Transportation Systems*, Oct. 2009, pp. 1–6. doi: 10.1109/ITSC.2009.5309697.
[17] M. R. Jeong, J.-Y. Kwak, J. E. Son, B. Ko, and J.-Y. Nam, “Fast Pedestrian Detection Using a Night Vision System for Safety Driving,” in *2014 11th International Conference on Computer Graphics, Imaging and Visualization*, Singapore, Aug. 2014, pp. 69–72. doi: 10.1109/CGiV.2014.25.
[18] M. Kieu, L. Berlincioni, L. Galteri, M. Bertini, A. D. Bagdanov, and A. Del Bimbo, “Robust pedestrian detection in thermal imagery using synthesized images,” *arXiv:2102.02005*, Feb. 2021.
[19] D. Ghose, S. M. Desai, S. Bhattacharya, D. Chakraborty, M. Fiterau, and T. Rahman, “Pedestrian Detection in Thermal Images using Saliency Maps,” *ArXiv190406859 Cs*, Apr. 2019, Accessed: Apr. 02, 2021. [Online]. Available: http://arxiv.org/abs/1904.06859
[20] R. Girshick, “Fast R-CNN,” *ArXiv150408083 Cs*, Sep. 2015.
[21] Y. Fang, K. Yamada, Y. Ninomiya, B. K. P. Horn, and I. Masaki, “A Shape-Independent Method for Pedestrian Detection With Far-Infrared Images,” *IEEE Trans. Veh. Technol.*, vol. 53, no. 6, pp. 1679–1697, Nov. 2004, doi: 10.1109/TVT.2004.834875.
[22] Fengliang Xu, Xia Liu, and K. Fujimura, “Pedestrian detection and tracking with night vision,” *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 1, pp. 63–71, Mar. 2005, doi: 10.1109/TITS.2004.838222.
[23] A. González *et al.*, “Pedestrian Detection at Day/Night Time with Visible and FIR Cameras: A Comparison,” *Sensors*, vol. 16, no. 6, p. 820, Jun. 2016, doi: 10.3390/s16060820.
[24] S.-S. Kim, I.-Y. Gwak, and S.-W. Lee, “Coarse-to-Fine Deep Learning of Continuous Pedestrian Orientation Based on Spatial Co-Occurrence Feature,” *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 6, pp. 2522–2533, Jun. 2020, doi: 10.1109/TITS.2019.2919920.
[25] J. H. Kim, H. G. Hong, and K. R. Park, “Convolutional Neural Network-Based Human Detection in Nighttime Images Using Visible Light Camera Sensors,” *Sensors*, vol. 17, no. 5, p. 1065, May 2017, doi: 10.3390/s17051065.
[26] S. Hwang, J. Park, N. Kim, Y. Choi, and I. S. Kweon, “Multispectral pedestrian detection: Benchmark dataset and baseline,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1037–1045. doi: 10.1109/CVPR.2015.7298706.
[27] D. Olmeda, C. Premebida, U. Nunes, J. M. Armingol, and A. de la Escalera, “Pedestrian detection in far infrared images,” *Integr. Comput.-Aided Eng.*, vol. 20, no. 4, pp. 347–360.
[28] P. Williams, K. H. Norris, and American Association of Cereal Chemists, Eds., *Near-infrared technology: in the agricultural and food industries*, 2nd ed. St. Paul, Minn: American Association of Cereal Chemists, 2001.
[29] B. Więcek *et al.*, *Termografia i spektrometria w podczerwieni: zastosowania przemysłowe*. Warszawa: Wydawnictwo WNT : Wydawnictwo Naukowe PWN, 2017.

[30] F. Jahard, “Far/near infrared adapted pyramid-based fusion for automotive night vision,” in *6th International Conference on Image Processing and its Applications*, Dublin, Ireland, 1997, vol. 1997, pp. 886–890. doi: 10.1049/cp:19971024.
[31] Y. Luo, J. Remillard, and D. Hoetzer, “Pedestrian detection in near-infrared night vision system,” in *2010 IEEE Intelligent Vehicles Symposium*, La Jolla, CA, USA, Jun. 2010, pp. 51–58. doi: 10.1109/IVS.2010.5548089.
[32] T. Kim and S. Kim, “Pedestrian detection at night time in FIR domain: Comprehensive study about temperature and brightness and new benchmark,” *Pattern Recognit.*, vol. 79, pp. 44–54, Jul. 2018, doi: 10.1016/j.patcog.2018.01.029.
[33] O. Tsimhoni, J. Bärgman, and M. J. Flannagan, “Pedestrian Detection with near and far Infrared Night Vision Enhancement,” *LEUKOS*, vol. 4, no. 2, pp. 113–128, Oct. 2007, doi: 10.1582/LEUKOS.2007.04.02.003.
[34] M. Vollmer and K.-P. Möllmann, *Infrared thermal imaging: fundamentals, research and applications*, Second edition. Weinheim, Germany: Wiley-VCH Verlag GmbH & Co. KGaA, 2018.
[35] *The Ultimate Infrared Handbook for R&D Professionals.* FLIR AB. [Online]. Available: https://www.flir.com/discover/rd-science/the-ultimate-infrared-handbook-for-rnd-professionals/
[36] H. Madura and W. Minkina, *Pomiary termowizyjne w praktyce: praca zbiorowa*. Warszawa: Agenda Wydawnicza PAKu, 2004.
[37] M. Bertozzi, A. Broggi, M. Felisa, G. Vezzoni, and M. Del Rose, “Low-level Pedestrian Detection by means of Visible and Far Infra-red Tetra-vision,” in *2006 IEEE Intelligent Vehicles Symposium*, Meguro-Ku, Japan, 2006, pp. 231–236. doi: 10.1109/IVS.2006.1689633.
[38] M. Teutsch, T. Mueller, M. Huber, and J. Beyerer, “Low Resolution Person Detection with a Moving Thermal Infrared Camera by Hot Spot Classification,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Jun. 2014, pp. 209–216. doi: 10.1109/CVPRW.2014.40.
[39] K. Park, S. Kim, and K. Sohn, “Unified multi-spectral pedestrian detection based on probabilistic fusion networks,” *Pattern Recognit.*, vol. 80, pp. 143–155, Aug. 2018, doi: 10.1016/j.patcog.2018.03.007.
[40] Y. Socarrás, S. Ramos, D. Vázquez, A. M. Lopez, and T. Gevers, “Adapting Pedestrian Detection from Synthetic to Far Infrared Images,” presented at the Conference: ICCV -- Workshop on Visual Domain Adaptation and Dataset Bias, 2013.
[41] Y. Zhang, Y. Zhao, and G. Li, “Grey self-similarity feature for night-time pedestrian detection,” *J. Comput. Inf. Syst.*, vol. 10, no. 7, pp. 2967–2974, 2014.
[42] J. W. Davis and M. A. Keck, “A Two-Stage Template Approach to Person Detection in Thermal Imagery,” in *2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION’05) - Volume 1*, Jan. 2005, vol. 1, pp. 364–369. doi: 10.1109/ACVMOT.2005.14.
[43] V. John, S. Mita, Z. Liu, and B. Qi, “Pedestrian detection in thermal images using adaptive fuzzy C-means clustering and convolutional neural networks,” in *2015 14th IAPR International Conference on Machine Vision Applications (MVA)*, May 2015, pp. 246–249. doi: 10.1109/MVA.2015.7153177.
[44] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele, “Towards Reaching Human Performance in Pedestrian Detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 973–986, Apr. 2018, doi: 10.1109/TPAMI.2017.2700460.

[45] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” *ArXiv150601497 Cs*, Jan. 2016, Accessed: Jun. 22, 2020. [Online]. Available: http://arxiv.org/abs/1506.01497
[46] J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement,” 2018.
[47] H. Sun, C. Wang, B. Wang, and N. El-Sheimy, “Pyramid binary pattern features for real-time pedestrian detection from infrared videos,” *Neurocomputing*, vol. 74, no. 5, pp. 797–804, Feb. 2011, doi: 10.1016/j.neucom.2010.10.009.
[48] D. Hutchison *et al.*, “Multi-stage Sampling with Boosting Cascades for Pedestrian Detection in Images and Videos,” in *Computer Vision – ECCV 2010*, vol. 6316, K. Daniilidis, P. Maragos, and N. Paragios, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 196–209. doi: 10.1007/978-3-642-15567-3_15.
[49] M. Pätzold, R. H. Evangelio, and T. Sikora, “Counting People in Crowded Environments by Fusion of Shape and Motion Information,” in *2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance*, Aug. 2010, pp. 157–164. doi: 10.1109/AVSS.2010.92.
[50] A. Broggi, A. Fascioli, I. Fedriga, A. Tibaldi, and M. D. Rose, “Stereo-based preprocessing for human shape localization in unstructured environments,” in *IEEE IV2003 Intelligent Vehicles Symposium. Proceedings (Cat. No.03TH8683)*, Jun. 2003, pp. 410–415. doi: 10.1109/IVS.2003.1212946.
[51] D. M. Gavrila, J. Giebel, and S. Munder, “Vision-based pedestrian detection: the PROTECTOR system,” in *IEEE Intelligent Vehicles Symposium, 2004*, Jun. 2004, pp. 13–18. doi: 10.1109/IVS.2004.1336348.
[52] R. Labayrade, D. Aubert, and J.- Tarel, “Real time obstacle detection in stereovision on non flat road geometry through ‘v-disparity’ representation,” in *IEEE Intelligent Vehicle Symposium, 2002*, Jun. 2002, vol. 2, pp. 646–651 vol.2. doi: 10.1109/IVS.2002.1188024.
[53] H. Elzein, S. Lakshmanan, and P. Watta, “A motion and shape-based pedestrian detection algorithm,” in *IEEE IV2003 Intelligent Vehicles Symposium. Proceedings (Cat. No.03TH8683)*, Jun. 2003, pp. 500–504. doi: 10.1109/IVS.2003.1212962.
[54] N. Dalal, B. Triggs, and C. Schmid, “Human Detection Using Oriented Histograms of Flow and Appearance,” in *Computer Vision – ECCV 2006*, Berlin, Heidelberg, 2006, pp. 428–441. doi: 10.1007/11744047_33.
[55] C. Dai, Y. Zheng, and X. Li, “Pedestrian detection and tracking in infrared imagery using shape and appearance,” *Comput. Vis. Image Underst.*, vol. 106, no. 2, pp. 288–299, May 2007, doi: 10.1016/j.cviu.2006.08.009.
[56] C. Li, D. Song, R. Tong, and M. Tang, “Illumination-aware faster R-CNN for robust multispectral pedestrian detection,” *Pattern Recognit.*, vol. 85, pp. 161–171, Jan. 2019, doi: 10.1016/j.patcog.2018.08.005.
[57] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” *ArXiv13112524 Cs*, Oct. 2014, Accessed: Apr. 02, 2021. [Online]. Available: http://arxiv.org/abs/1311.2524
[58] U. Meis, W. Ritter, and H. Neumann, “Detection and classification of obstacles in night vision traffic scenes based on infrared imagery,” in *Proceedings of the 2003 IEEE International Conference on Intelligent Transportation Systems*, Shanghai, China, 2003, vol. 2, pp. 1140–1144. doi: 10.1109/ITSC.2003.1252663.
[59] H. Nanda and L. Davis, “Probabilistic template based pedestrian detection in infrared videos,” in *Intelligent Vehicle Symposium, 2002. IEEE*, Versailles, France, 2003, vol. 1, pp. 15–20. doi: 10.1109/IVS.2002.1187921.

[60] N. Otsu, “A Threshold Selection Method from Gray-Level Histograms,” *IEEE Trans. Syst. Man Cybern.*, vol. 9, no. 1, pp. 62–66, Jan. 1979, doi: 10.1109/TSMC.1979.4310076.
[61] M. Bertozzi, A. Broggi, A. Lasagni, and M. D. Rose, “Infrared stereo vision-based pedestrian detection,” in *IEEE Proceedings. Intelligent Vehicles Symposium, 2005.*, Las Vegas, NV, USA, 2005, pp. 24–29. doi: 10.1109/IVS.2005.1505072.
[62] M. Bertozzi, A. Broggi, C. Caraffi, M. Del Rose, M. Felisa, and G. Vezzoni, “Pedestrian detection by means of far-infrared stereo vision,” *Comput. Vis. Image Underst.*, vol. 106, no. 2–3, pp. 194–204, May 2007, doi: 10.1016/j.cviu.2006.07.016.
[63] M. Bertozzi, A. Broggi, A. Fascioli, T. Graf, and M. Meinecke, “Pedestrian Detection for Driver Assistance Using Multiresolution Infrared Vision,” *IEEE Trans. Veh. Technol.*, vol. 53, no. 6, pp. 1666–1678, Nov. 2004, doi: 10.1109/TVT.2004.834878.
[64] D. Olmeda, C. Hilario, A. de la Escalera, and J. M. Armingol, “Pedestrian Detection and Tracking Based on Far Infrared Visual Information,” in *Advanced Concepts for Intelligent Vision Systems*, Berlin, Heidelberg, 2008, pp. 958–969.
[65] D. Olmeda, J. M. Armingol, and A. de la Escalera, “Discrete features for rapid pedestrian detection in infrared images,” in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura-Algarve, Portugal, Oct. 2012, pp. 3067–3072. doi: 10.1109/IROS.2012.6385928.
[66] U. Meis, M. Oberldrider, and W. Ritter, “Reinforcing the reliability of pedestrian detection in far-infrared sensing,” in *IEEE Intelligent Vehicles Symposium, 2004*, Parma, Italy, 2004, pp. 779–783. doi: 10.1109/IVS.2004.1336483.
[67] X. Xu, S. Xu, L. Jin, and E. Song, “Characteristic analysis of Otsu threshold and its applications,” *Pattern Recognit. Lett.*, vol. 32, no. 7, pp. 956–961, May 2011, doi: 10.1016/j.patrec.2011.01.021.
[68] K. Piniarski, P. Pawłowski, and A. Dąbrowski, “Video Processing Algorithms for Detection of Pedestrians,” *Comput. Methods Sci. Technol. CMST*, vol. 21, no. 3, pp. 141–150, 2015, doi: 10.12921/CMST.2015.21.03.005.
[69] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, San Diego, CA, USA, 2005, vol. 1, pp. 886–893. doi: 10.1109/CVPR.2005.177.
[70] Y. Cao, S. Pranata, and H. Nishimura, “Local Binary Pattern features for pedestrian detection at night/dark environment,” in *2011 18th IEEE International Conference on Image Processing*, Brussels, Belgium, Sep. 2011, pp. 2053–2056. doi: 10.1109/ICIP.2011.6115883.
[71] S. Zhang, C. Bauckhage, and A. B. Cremers, “Efficient Pedestrian Detection via Rectangular Features Based on a Statistical Shape Model,” *IEEE Trans. Intell. Transp. Syst.*, pp. 1–13, 2014, doi: 10.1109/TITS.2014.2341042.
[72] Y. Wei, Q. Tian, and T. Guo, “An Improved Pedestrian Detection Algorithm Integrating Haar-Like Features and HOG Descriptors,” *Adv. Mech. Eng.*, vol. 5, p. 546206, Jan. 2013, doi: 10.1155/2013/546206.
[73] L. C. Padierna, M. Carpio, A. Rojas-Domínguez, H. Puga, and H. Fraire, “A novel formulation of orthogonal polynomial kernel functions for SVM classifiers: The Gegenbauer family,” *Pattern Recognit.*, vol. 84, pp. 211–225, Dec. 2018, doi: 10.1016/j.patcog.2018.07.010.
[74] M. Bilal and M. S. Hanif, “Benchmark Revision for HOG-SVM Pedestrian Detector Through Reinvigorated Training and Evaluation Methodologies,” *IEEE*

Trans. Intell. Transp. Syst., vol. 21, no. 3, pp. 1277–1287, Mar. 2020, doi: 10.1109/TITS.2019.2906132.
[75] P. Dollar, Z. Tu, and P. Perona, "Integral Channel Features," presented at the British Machine Vision Conference, 2009.
[76] P. Dollar, R. Appel, S. Belongie, and P. Perona, "Fast Feature Pyramids for Object Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1532–1545, Aug. 2014, doi: 10.1109/TPAMI.2014.2300479.
[77] S. Zhang, R. Benenson, and B. Schiele, "Filtered channel features for pedestrian detection," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 1751–1760. doi: 10.1109/CVPR.2015.7298784.
[78] Y. Jia *et al.*, "Caffe: Convolutional Architecture for Fast Feature Embedding," *ArXiv14085093 Cs*, Jun. 2014, Accessed: Jun. 20, 2020. [Online]. Available: http://arxiv.org/abs/1408.5093
[79] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.
[80] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *ArXiv14091556 Cs*, Apr. 2015, Accessed: Jun. 20, 2020. [Online]. Available: http://arxiv.org/abs/1409.1556
[81] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *ArXiv151203385 Cs*, Dec. 2015, Accessed: Jun. 20, 2020. [Online]. Available: http://arxiv.org/abs/1512.03385
[82] P. Pawłowski, K. Piniarski, and A. Dąbrowski, "Pedestrian detection in low resolution night vision images," in *2015 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, Poznan, Poland, Sep. 2015, pp. 185–190. doi: 10.1109/SPA.2015.7365157.
[83] P. Viola and M. J. Jones, "Robust Real-Time Face Detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, May 2004, doi: 10.1023/B:VISI.0000013087.49260.fb.
[84] L. Guo, P.-S. Ge, M.-H. Zhang, L.-H. Li, and Y.-B. Zhao, "Pedestrian detection for intelligent transportation systems combining AdaBoost algorithm and support vector machine," *Expert Syst. Appl.*, vol. 39, no. 4, pp. 4274–4286, Mar. 2012, doi: 10.1016/j.eswa.2011.09.106.
[85] Z. Wang, S. Yoon, S. J. Xie, Y. Lu, and D. S. Park, "A High Accuracy Pedestrian Detection System Combining a Cascade AdaBoost Detector and Random Vector Functional-Link Net," *Sci. World J.*, vol. 2014, pp. 1–7, 2014, doi: 10.1155/2014/105089.
[86] K.-K. Kong and K.-S. Hong, "Design of coupled strong classifiers in AdaBoost framework and its application to pedestrian detection," *Pattern Recognit. Lett.*, vol. 68, pp. 63–69, Dec. 2015, doi: 10.1016/j.patrec.2015.07.043.
[87] K. Piniarski and P. Pawłowski, "Multi-branch classifiers for pedestrian detection from infrared night and day images," in *2016 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, Sep. 2016, pp. 248–253. doi: 10.1109/SPA.2016.7763622.
[88] S. P. Jeon, Y. S. Lee, and K. N. Choi, "Movement direction-based approaches for pedestrian detection in road scenes," in *2015 21st Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV)*, Jan. 2015, pp. 1–4. doi: 10.1109/FCV.2015.7103727.

[89] Y.-L. Hou, Y. Song, X. Hao, Y. Shen, M. Qian, and H. Chen, “Multispectral pedestrian detection based on deep convolutional neural networks,” *Infrared Phys. Technol.*, vol. 94, pp. 69–77, Nov. 2018, doi: 10.1016/j.infrared.2018.08.029.
[90] Ming-Kuei Hu, “Visual pattern recognition by moment invariants,” *IEEE Trans. Inf. Theory*, vol. 8, no. 2, pp. 179–187, Feb. 1962, doi: 10.1109/TIT.1962.1057692.
[91] F. Chaumette, “Image Moments: A General and Useful Set of Features for Visual Servoing,” *IEEE Trans. Robot.*, vol. 20, no. 4, pp. 713–723, Aug. 2004, doi: 10.1109/TRO.2004.829463.
[92] P. Dollar, C. Wojek, B. Schiele, and P. Perona, “Pedestrian Detection: An Evaluation of the State of the Art,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, Apr. 2012, doi: 10.1109/TPAMI.2011.155.
[93] C. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer-Verlag, 2006. Accessed: Jun. 20, 2020. [Online]. Available: https://www.springer.com/gp/book/9780387310732
[94] V. Vapnik, *The Nature of Statistical Learning Theory*, 2nd ed. New York: Springer-Verlag, 2000. doi: 10.1007/978-1-4757-3264-1.
[95] “Open source computer vision – OpenCV,” 2019. http://opencv.org
[96] C.-C. Chang and C.-J. Lin, “LIBSVM: A library for support vector machines,” *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, Apr. 2011, doi: 10.1145/1961189.1961199.
[97] “Keras: The Python Deep Learning library,” 2019. https://keras.io
[98] “NuGet Gallery | Home.” https://www.nuget.org/ (accessed Aug. 02, 2021).
[99] J. Cao, Y. Pang, J. Xie, F. S. Khan, and L. Shao, “From Handcrafted to Deep Features for Pedestrian Detection: A Survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2021, doi: 10.1109/TPAMI.2021.3076733.
[100] J. U. Kim, S. Park, and Y. M. Ro, “Uncertainty-Guided Cross-Modal Learning for Robust Multispectral Pedestrian Detection,” *IEEE Trans. Circuits Syst. Video Technol.*, pp. 1–1, 2021, doi: 10.1109/TCSVT.2021.3076466.
[101] D. Xu, W. Ouyang, E. Ricci, X. Wang, and N. Sebe, “Learning Cross-Modal Deep Representations for Robust Pedestrian Detection,” *ArXiv170402431 Cs*, Jan. 2018, Accessed: Jun. 16, 2021. [Online]. Available: http://arxiv.org/abs/1704.02431
[102] K. Zhou, L. Chen, and X. Cao, “Improving Multispectral Pedestrian Detection by Addressing Modality Imbalance Problems,” Aug. 2020, Accessed: Jun. 11, 2021. [Online]. Available: https://arxiv.org/abs/2008.03043v2
[103] L. Zhang, X. Zhu, X. Chen, X. Yang, Z. Lei, and Z. Liu, “Weakly Aligned Cross-Modal Learning for Multispectral Pedestrian Detection,” in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South), Oct. 2019, pp. 5126–5136. doi: 10.1109/ICCV.2019.00523.
[104] M. Kanmani and V. Narasimhan, “An optimal weighted averaging fusion strategy for thermal and visible images using dual tree discrete wavelet transform and self tunning particle swarm optimization,” *Multimed. Tools Appl.*, vol. 76, no. 20, pp. 20989–21010, Oct. 2017, doi: 10.1007/s11042-016-4030-x.
[105] K. Strandemar and M. Ingerhed, “Camera and method for use with camera,” US7544944B2, 2007 [Online]. Available: https://patents.google.com/patent/US7544944