Article

Semi-Supervised Faster RCNN-Based Person Detection and Load Classification for Far Field Video Surveillance

Haoran Wei * and Nasser Kehtarnavaz

Department of Electrical and Computer Engineering, University of Texas at Dallas, Richardson 75080, USA

* Correspondence: Haoran.Wei@utdallas.edu; Tel.: +1-972-883-6838

Received: 26 May 2019; Accepted: 25 June 2019; Published: 27 June 2019

Abstract: This paper presents a semi-supervised faster region-based convolutional neural network (SF-RCNN) approach to detect persons and to classify the load carried by them in video data captured from distances several miles away via high-power lens video cameras. For detection, a set of computationally efficient image processing steps are considered to identify moving areas that may contain a person. These areas are then passed onto a faster RCNN classifier whose convolutional layers consist of ResNet50 transfer learning. Frame labels are obtained in a semi-supervised manner for the training of the faster RCNN classifier. For load classification, another convolutional neural network classifier whose convolutional layers consist of GoogleNet transfer learning is used to distinguish a person carrying a bundle from a person carrying a long arm. Despite the challenges associated with the video dataset examined in terms of the low resolution of persons, the presence of heat haze, and the shaking of the camera, it is shown that the developed approach outperforms the faster RCNN approach.

Keywords: person detection and load classification; semi-supervised faster RCNN; far field video surveillance

1. Introduction and Related Works

The video surveillance market is currently valued at more than \$35 billion and is estimated to grow to more than \$65 billion in five years [1]. There are many applications of video surveillance, for example, traffic monitoring, public safety, parking lot monitoring, theft detection in the retail industry, and crime prevention. Many image processing algorithms for detection of a specific object or event in video data have been developed in the literature. Recently, the use of deep learning in image processing has experienced tremendous growth and deep learning approaches have been applied to video surveillance applications. This growth has occurred as, in many image processing applications, deep learning solutions have outperformed conventional solutions where detection/recognition is normally performed based on some handcrafted features that are designed for a specific application. When using deep learning approaches, the design of handcrafted features is not needed and the raw image data can be fed directly into a deep learning network to achieve detection/recognition.

Person detection for the application of pedestrian monitoring has been well studied. Three noteworthy pedestrian detection algorithms are reported in the literature [2–4]. In the work of [2], histograms of oriented gradients (HOG) features together with a support vector machine (SVM) classifier were used. In the work of [3], integral channel features (ICF) together with an AdaBoost classifier were used. In the work of [4], aggregated channel features (ACF) with an AdaBoost classifier were used. Variations of these methods have appeared in the literature [5–11]. More recently, convolutional neural network (CNN)-based approaches have shown improvements over conventional

approaches in pedestrian detection. These deep learning-based pedestrian detection approaches involve either a two-stage or a single-stage approach. Examples of two-stage approaches include region-based convolutional neural network (RCNN) [12], fast RCNN [13], and faster RCNN [14]. These approaches perform both region scanning and detection. Although, in general, these approaches have higher accuracy, their computational complexity is higher as well. Examples of single-stage approaches include SSD (single shot detector) [15] and the YOLO (you only look once) [16]. These approaches do not address region scanning. In general, these approaches have higher computational efficiency, but lower accuracy. Variations of the above deep learning-based approaches have also appeared in the literature [17–21].

For the far field video surveillance application, the use of these methods poses challenges owing to the lack of a large dataset and the low resolution of images involved. Because of the lack of a large dataset, a semi-supervised faster RCNN (SF-RCNN) approach is developed in this paper to achieve person detection and load classification based on far field video data. Far field indicates the use of high-power lenses to enable monitoring at distances that are three to five miles away. The application of interest here for far field video surveillance involves monitoring borders from a far distance for illegal crossing or activities. More specifically, the application of interest involves monitoring borders from a far distance in order to detect persons and to identify the load they carry. The loads of interest include drug bundles and long arms.

A two-stage approach is developed in this paper to address both person detection and load classification based on far field video data. During the person detection stage, a fast or computationally efficient approach for detecting moving areas in an image is considered, followed by a person detector to see whether there is a person in the moving areas. This approach is compared with the state-of-the-art faster RCNN person detector. The developed person detector is first trained by the Caltech pedestrian detection dataset, and then re-trained in a semi-supervised manner by the unlabeled far field video dataset. During the classification stage, a CNN with transfer learning is used to distinguish between the situations involving a person carrying a bundle and a person carrying a long arm. In our previous work [22], the detection was done using the AdaBoost person detector, while in this paper, the detection is carried out using the developed SF-RCNN person detector.

The rest of the paper is organized as follows. A description of the far field video dataset is provided in Section 2. The architectures of the person detection and load classification deep learning networks used are covered in Section 3. The experimental results and their discussion are then reported in Section 4. Finally, the conclusion is stated in Section 5.

2. Far Field Video Dataset

A dataset of far field video clips was made available for this work by the company Elbit Systems of America. The dataset consists of 32 video clips at 30 frames per second. Seventeen of the video clips were labeled as ‘Bundle’ video clips and 15 as ‘Long Arm’ video clips, denoting the loads carried by the person in the video clip. Figure 1 provides two sample images in these video clips, which are (1080 × 1920) pixels in size. Figure 1a corresponds to a person carrying a long arm, and Figure 1b corresponds to a person carrying a bundle. A zoomed version of the area in which a person was detected is also shown on the right side of these figures. The video clips were captured from a three-mile distance. No frame-level labels are provided for these video clips, meaning that it is unknown when a person will appear in the scene.

It is worth stating that the video data for this far field video surveillance application differ in appearance from the video data often seen for the pedestrian detection application. Far field video data involve the following challenges that do not appear in pedestrian monitoring video data: (1) as the video is taken from a far distance, a person appears in the scene with low resolution and, in many cases, with only the upper portion of the body being visible; (2) the presence of noise generated by the shaking of the camera due to wind or going out of focus; and (3) the presence of noise generated by heat haze as a result of the distance being far.

Besides the far field video dataset, the Caltech pedestrian dataset [23] is also considered here for training the person detection model. The Caltech pedestrian dataset consists of approximately 10 h of (640×480) 30 Hz video taken from a vehicle driving through regular traffic in an urban environment, providing about 250,000 frames with a total of 350,000 bounding boxes and 2300 pedestrian annotations. The details of this dataset appear in the work of [23].

Figure 1. Sample video clip images: (a) a person carrying a long arm and (b) a person carrying a bundle.

3. Developed Detection and Classification Approach

The steps involved in the developed person detection and load classification are illustrated in Figure 2. Initially, it is required to detect or locate the presence of a person in an entire image before performing any classification. Because of the large size of the image frames (1080×1920), it was found to be computationally inefficient to apply a person detector algorithm to the entire image. To make the detection process computationally efficient, a moving or changing areas detection step is first considered. The person detector is then applied only to the moving or changing areas and thus not to the entire image. Next, the detected area or sub-image in which a person is detected is passed onto a load classifier to obtain the load carried by the detected person. In addition to a frame-level classification, a video-level label is generated based on the image or frame-level labels.

Figure 2. Steps involved in the developed detection and classification approach.

3.1. Moving Areas Detection

A simple moving areas detection is applied first to allow the person detection module to operate in a computationally efficient manner. Although there are many background subtraction methods that can be used to find moving areas—for example, see the literature [24–33]—a simple frame differencing is utilized here to provide the input to the deep learning-based person detector. The person detector

then corrects remaining errors associated with moving areas. Note that because the camera is located miles away, camera shaking and heat haze make the background unstable, and background subtraction alone would not lead to a robust outcome.

To detect moving areas in an image, the steps illustrated in Figure 3 are considered. First, the image frames are down-sampled five times to (216×384) , so that the computational efficiency of the subsequent processing steps or components is increased. Then, the captured RGB images are converted into one luminance or gray-scale image using the luminance equation $Y = (R + G + B)/3$. Next, the difference of consecutive frames is passed through a convolution operation with an averaging filter to obtain the most significant moving or changing area in the image in a computationally efficient manner. It is worth noting that camera shaking due to winds also leads to the detection of moving areas. No camera stabilization is applied as part of our processing pipeline as this would add a considerable amount of computation time, not allowing our detection and classification solution to run in real-time on a regular computer. Sample outcomes of the color to luminance conversion and frame differencing steps are shown in Figure 4. Note that detected moving areas may occur because of heat haze noise or the presence of moving objects other than persons such as animals or cars. Figure 4 also includes images corresponding to detected moving areas, one with and one without a person in it.

Figure 3. Steps in moving areas detection.

Figure 4. Sample images: (a) luminance, (b) frame differencing, (c) detected moving area containing a moving person, (d) detected moving area containing no moving person caused by heat haze.

3.2. Person Detection

The next step of the approach consists of passing the detected moving areas to a person detector to generate boxes around the person in the scene. In our previous work [22], the person detection was done using the AdaBoost person detector. AdaBoost, short for adaptive boosting [34], involves forming a classifier as a linear combination of simple classifiers. In the work of [4], it was shown that the ACF features together with an AdaBoost classifier performed better than the HOG features together with an SVM classifier. Furthermore, in the work of [35], it was shown that the deep learning-based RCNN approach outperformed the AdaBoost approach for the pedestrian detection application.

3.2.1. Faster RCNN Detector

Faster RCNN is an extension of the RCNN and fast RCNN networks that have been used for object detection applications, which are variations of the CNN network. The main difference between them is how regions get selected for processing. RCNN and fast RCNN use a region selection algorithm such as Edge Boxes [36] or Selective Search [37], which are independent of the CNN network. Faster RCNN does the region selection as part of the CNN training and detection.

To address the limited amount of training data, the transfer learning method is considered here. In transfer learning, pre-trained CNN models are used. These pre-trained models are trained using big datasets. The layers of the pre-trained models are used up to the last fully connected layer. The last fully connected layer is trained using the dataset associated with this application. More details of the transfer learning method appear in the work of [38]. Here, the transfer learning method based on the pre-trained ResNet50 [39] model is used.

ResNet50 is a convolutional neural network that is trained on more than a million images from the ImageNet database [40]. The ImageNet database consist of 1.2 million images classified into 1000 classes. A block diagram illustrating the ResNet50 transfer learning architecture is shown in Figure 5. As illustrated in this figure, the ResNet50 architecture consists of convolution layers with skip layer connections, average pooling layers, and fully connected layers of processing elements. The details of these layers are discussed in the work of [39].

Figure 5. Diagram illustrating the ResNet-50 transfer learning architecture used.

3.2.2. Semi-Supervised Faster RCNN (SF-RCNN) Detector

Semi-supervised learning is a machine learning approach that makes use of both labeled and unlabeled data for training. It starts with a model trained using labeled data and then improves the performance using unlabeled data. As manual labeling is time consuming and labor intensive, the semi-supervised approach makes the training process more efficient. More details regarding semi-supervised learning are described in the work of [41].

Noting that the far field video dataset does not give frame-level labels, the supervised training is first done using the Caltech pedestrian dataset. When the model is tested using the far field video dataset, one faces a mismatch between the training and testing datasets. To address this mismatch, the semi-supervised method is adopted in order to first obtain frame-level labels automatically from the unlabeled far field video dataset using a “high threshold” (e.g., 0.99) for person detection in the Caltech pedestrian dataset. Then, the automatically labeled persons in the far field data are used to further train the faster RCNN network. This training process allows the deep learning model to learn the common features in both the Caltech pedestrian and far field datasets. During testing or operation, a “nominal threshold” (e.g., 0.6) is used to ensure that all persons get detected for the load classification stage. A block diagram illustrating the developed semi-supervised architecture is shown in Figure 6.

Figure 6. Diagram illustrating the developed semi-supervised learning architecture.

3.3. Load Classification

After identifying areas or sub-images in which a person is present, these areas or sub-images are passed to a CNN classifier. Considering that during the detection stage, misdetection could occur, areas that contain trees, grass, or other objects were manually extracted and placed into a third class labeled ‘Others’. Also, background areas from two of the Bundle video clips were randomly selected and were placed into the ‘Others’ class manually. These two video clips were thus not used in the

experimentations reported in Section 4. In other words, 30 video clips (15 Bundle and 15 Long Arm) were used for the training and testing of the CNN classifier, outputting three classes consisting of person with long arm, person with bundle, and others. The leave-one-out cross validation technique was carried out, that is, 29 video clips were used for training and the remaining video clip was used for testing. The training and testing was repeated 30 times, each time selecting a different video clip for testing and a different set of 29 video clips for training. The results were averaged over the 30 repetitions of training and testing.

Again, because of the lack of a large dataset, transfer learning is adopted during this stage as well. The transfer learning method based on the pre-trained AlexNet [42] model, the pre-trained GoogleNet [43], and the pre-trained ResNet50 [39] model were considered. Four CNN approaches were thus examined. The first approach was self-defined CNN, meaning that the training was done using the dataset described earlier. The self-defined CNN model included three convolution layers, two max-pooling layers, and three fully connected layers. The second, third, and fourth CNN approaches incorporated pre-trained models of AlexNet, GoogleNet, and ResNet50, respectively. These networks were trained using the ImageNet database [40].

The classification above was done on a per image basis. It is also possible to carry out the classification on a per video clip basis by majority voting. For video-based classification, there are only two classes. For image-based classification, the third class ‘Others’ was manually created, which is not applicable to the video-based classification.

4. Experimental Results and Discussion

In this section, the experimental results of the developed person detection and load classification approach are reported. First, the detection and classification were examined separately, and then the detection and classification were evaluated together. The results corresponding to the real-time aspect of the developed approach are also provided. All of the coding was done in MATLAB 2018b and the timing results reported are for a personal computer equipped with an Intel i7-7700K CPU (central processing unit) and an NVIDIA QuadroP4000 GPU (graphics processing unit).

4.1. Person Detection Results

The developed SF-RCNN approach was compared to the faster RCNN, which is increasingly being used for person detection. The MATLAB faster RCNN training function [44] was used for the faster RCNN and SF-RCNN. This function includes a so-called region proposal network (RPN), an ROI (region of interest) max pooling layer, and classification and regression layers. The performance metrics used include FPR (false positive rate), TPR (true positive rate), and FNR (false negative rate), which are widely used and are defined as follows:

$$FPR = \frac{FP}{FP + TN}, \quad (1)$$

$$TPR = \frac{TP}{TP + FN}, \quad (2)$$

$$FNR = \frac{FN}{TP + NR}, \quad (3)$$

where TP , FP , FN , and FP denote the number of true positives, the number of false positives, the number of false negatives, and the number of false positives, respectively. In our entire dataset of 30 video clips, there were a total of 65,220 frames, with 30,060 frames as true positive or denoting the presence of a person, and 35,160 frames as true negative or denoting the absence of a person.

When using an RCNN, a score gets generated for each detecting box that can be thresholded. Table 1 shows the results for the faster RCNN and SF-RCNN approaches at different thresholds. Lower thresholds led to more detected boxes as well as higher errors. An ROC (receiver operating

characteristic) curve was plotted based on TPR and FPR (see Figure 7), indicating the performance at different thresholds. As can be seen from Table 1 and Figure 7, the SF-RCNN approach generated a higher ROC curve.

Table 1. Detection performance. FPR—false positive rate; TPR—true positive rate; FNR—false negative rate; RCNN—region-based convolutional neural network.

Approach	Threshold	FPR	TPR	FNR
Faster RCNN	0.95	0.006%	21.9%	78.06%
	0.6	0.03%	34.06%	64.94%
	0.01	3.40%	50.83%	49.17%
	0.001	25.71%	55.47%	44.53%
	0.95	0.003%	47.42%	52.58%
Semi-Supervised Faster RCNN	0.6	0.006%	51.73%	48.27%
	0.01	1.28%	55.21%	44.79%
	0.001	6.64%	56.92%	43.08%
	0.0001	15.93%	59.16%	40.84%

Figure 7. Receiver operating characteristic (ROC) curves of the semi-supervised faster region-based convolutional neural network (SF-RCNN) versus faster RCNN approaches for person detection.

The processing time for both the faster RCNN and SF-RCNN person detection was 6.5 s when using the CPU, which dropped to 0.4 s per image frame when using the GPU. These times included the time for reading image frames.

4.2. Load Classification Results

For the classification component, four CNN approaches were considered. The images used consisted of the areas with known manually identified class labels. Figure 8 provides the average accuracy of the four different CNN approaches examined, as well as the processing times when using the CPU and when using the GPU. From Figure 8, it can be seen that the GoogleNet and ResNet-50 transfer learning network provided the highest accuracy of 88%, with the GoogleNet having a lower

processing time. Also, the figure shows the speed up in the processing time per image frame when using the GPU instead of the CPU.

Figure 8. Different CNN approaches for classification.

The examination of the misclassifications indicated that they were primarily the result of persons with a long arm getting labeled as persons with a bundle, because in most of the long arm video clips, the persons also carried a bundle. As a result, when an image reflected the back or side of a person with a long arm, it resembled a person carrying a bundle.

4.3. Combined Detection and Classification Results

The results of our combined detection and classification are reported here, which is the way an actual system operates in the field. Image areas from the developed SF-RCNN person detector were used for training GoogleNet transfer learning without manually selecting labels. The confusion matrices of the combined detection and classification appear in Table 2. As can be seen from this table, for the image-based approach, an overall accuracy of 90.9% was obtained. More specifically, the developed person detector processed 15,553 images in the entire video dataset, consisting of 15,551 true positives images and 2 false positive images with a threshold of 0.6. For the total of 1415 mislabeled images, only 2 of them (0.14%) were due to the person detector and 1413 of them (99.86%) were due to the load classifier. In essence, the errors were nearly all caused by the classifier corresponding to the situations when the back or side of a person faced the camera.

Table 2. Confusion matrix of image-based combined detection and classification.

Identified Class True Class	Long Arm	Bundle
Long Arm	91.0%	9.0%
Bundle	9.3%	90.7%

A majority vote was taken over the image frames of a video clip to classify that video clip. This way, that is, for the video-based approach, the overall accuracy was found to be 93.3%. It was noticed that in two of the long arm video clips, the person's back was facing the camera. By not considering these two video clips, the overall accuracy for the video-based approach reached 100%.

4.4. Real-Time Processing

As noted earlier, the SF-RCNN person detection takes 0.4 s per image and the transfer learning GoogleNet takes 7 ms when using the GPU. This allows processing 2 frames per second when using a personal computer without any other image processing hardware board. A real-time processing operation was conducted by performing the detection and classification every half second or one per 15 frames. The average accuracy of the video-based approach when selecting different frames for the majority voting was found to be 94.4%. The confusion matrix of the real-time video-based approach for the combined detection and classification appears in Table 3.

Table 3. Confusion matrix of real-time video-based combined detection and classification.

Identified Class True Class	Long Arm	Bundle
Long Arm	92%	8%
Bundle	3.1%	96.9%

5. Conclusions

A semi-supervised faster RCNN approach was developed in this paper for the purpose of detecting persons and the load carried by them in far field video surveillance data that are captured at distances several miles away via a video camera fitted with a high-power lens. This approach was compared to the faster RCNN approach as the current state-of-the-art, and the results obtained indicated that the developed approach provides an effective solution for detecting and distinguishing a person carrying a bundle from a person carrying a long arm in the far field video dataset examined.

Possible future improvements include developing a dedicated hardware platform to run the processing pipeline at a higher frame rate and collecting more data for the training of the deep neural networks. A dedicated hardware platform would allow running computationally intensive and advanced preprocessing algorithms, such as image stabilization and background subtraction, in real-time as part of the detection and classification processing pipeline.

Author Contributions: H.W. and N.K. contributed equally to this work.

Funding: This work was funded by Elbit Systems of America through a contract with the University of Texas at Dallas.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Markets and Markets. Available online: <https://www.marketsandmarkets.com/Market-Reports/video-surveillance-market-645.html> (accessed on 20 February 2019).
2. Dalal, N.; Triggs, B. Histograms of Oriented Gradients for Human Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–25 June 2005; pp. 886–893.
3. Dollar, P.; Wojek, C.; Shiele, B.; Perona, P. Pedestrian Detection: A Benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 304–311.
4. Dollar, P.; Appel, R.; Belongie, S.; Perona, P. Fast Feature Pyramids for Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 1532–1545. [CrossRef] [PubMed]
5. Jiang, Y.; Wang, J.; Liang, Y.; Xia, J. Combining static and dynamic features for real-time moving pedestrian detection. *Multimed. Tools Appl.* **2019**, *78*, 3781–3795. [CrossRef]
6. Xiao, F.; Liu, B.; Li, R. Pedestrian object detection with fusion of visual attention mechanism and semantic computation. *Multimed. Tools Appl.* **2019**, 1–15. [CrossRef]

7. Hong, G.S.; Kim, B.G.; Hwang, Y.S.; Kwon, K.K. (2016) Fast multi-feature pedestrian detection algorithm based on histogram of oriented gradient using discrete wavelet transform. *Multimed. Tools Appl.* **2016**, *75*, 15229–15245. [[CrossRef](#)]
8. Yang, Y.; Liu, W.; Wang, Y.; Cai, Y. Research on the algorithm of pedestrian recognition in front of the vehicle based on SVM. In Proceedings of the 11th International Symposium on Distributed Computing and Applications to Business, Engineering and Science, DCABES 2012, Guilin, China, 19–22 October 2012; pp. 396–400.
9. Chavez-Garcia, R.O.; Aycard, O. Multiple Sensor Fusion and Classification for Moving Object Detection and Tracking. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 525–534. [[CrossRef](#)]
10. Wang, X.; Han, T.X.; Yan, S. An HOG-LBP human detector with partial occlusion handling. In Proceedings of the IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 32–39.
11. Roncancio, H.; Hernandez, A.C.; Becker, M. Vision-based system for pedestrian recognition using a tuned SVM classifier. In Proceedings of the Workshop on Engineering Applications, Bogota, Columbia, 2–4 May 2012.
12. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
13. Girshick, R. Fast R-CNN. In Proceedings of the International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
14. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Intell. Transp. Syst.* **2016**, *39*, 1137–1149. [[CrossRef](#)] [[PubMed](#)]
15. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 21–37.
16. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
17. Song, H.; Choi, I.K.; Ko, M.S.; Bae, J.; Kwak, S.; Yoo, J. Vulnerable pedestrian detection and tracking using deep learning. In Proceedings of the 2018 International Conference on Electronics, Information, and Communication (ICEIC), Honolulu, HI, USA, 24–27 January 2018; pp. 1–2.
18. Hou, Y.L.; Song, Y.; Hao, X.; Shen, Y.; Qian, M. Multispectral pedestrian detection based on deep convolutional neural networks. In Proceedings of the IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), Xiamen, China, 22–25 October 2017.
19. González, A.; Fang, Z.; Socarras, Y.; Serrat, J.; Vázquez, D.; Xu, J.; López, A.M. Pedestrian Detection at Day/Night Time with Visible and FIR Cameras: A Comparison. *Sensors* **2016**, *16*, 820. [[CrossRef](#)] [[PubMed](#)]
20. Hosang, J.; Benenson, R.; Dollar, P.; Schiele, B. What Makes for Effective Detection Proposals? *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 814–830. [[CrossRef](#)] [[PubMed](#)]
21. Brazil, G.; Yin, X.; Liu, X. Illuminating Pedestrians via Simultaneous Detection and Segmentation. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4960–4969.
22. Wei, H.; Laszewski, M.; Kehtarnavaz, N. Deep Learning-Based Person Detection and Classification for Far Field Video Surveillance. In Proceedings of the 13th IEEE Dallas Circuits and Systems Conference, Dallas, TX, USA, 2–12 November 2018; pp. 1–4.
23. Dollár, P.; Wojek, C.; Schiele, B.; Perona, P. Pedestrian detection: An evaluation of the state of the art. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *34*, 743–761. [[CrossRef](#)] [[PubMed](#)]
24. Bouwmans, T. Traditional and recent approaches in background modeling for foreground detection: An overview. *Comput. Sci. Rev.* **2014**, *11*, 31–66. [[CrossRef](#)]
25. Stauffer, C.; Grimson, W.E.L. Adaptive background mixture models for real-time tracking. In Proceedings of the 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Fort Collins, CO, USA, 23–25 June 1999; Volume 2.
26. Elgammal, A.; Harwood, D.; Davis, L. Non-Parametric Model for Background Subtraction. In *Computer Vision-ECCV 2000*; Springer: Berlin, Germany, 2000; pp. 751–767.
27. Heikkilä, M.; Pietikäinen, M.; Heikkilä, J. A texture-based method for detecting moving objects. In Proceedings of the British Machine Vision Conference (BMVC), Kingston, UK, 7–9 September 2004; pp. 1–10.

28. Yoshinaga, S.; Shimada, A.; Nagahara, H.; Taniguchi, R. Statistical Local Difference Pattern for Background Modeling. *IPSJ Trans. Comput. Vis. Appl.* **2011**, *3*, 198–210. [[CrossRef](#)]
29. Sultana, M.; Mahmood, A.; Javed, S.; Jung, S.K. Unsupervised Deep Context Prediction for Background Estimation and Foreground Segmentation. *Mach. Vision Appl.* **2019**, *30*, 375–395. [[CrossRef](#)]
30. Minematsu, T.; Shimada, A.; Uchiyama, H.; Taniguchi, R.I. Analytics of Deep Neural Network-based Background Subtraction. *J. Imaging* **2018**, *4*, 78. [[CrossRef](#)]
31. Bouwmans, T.; Javed, S.; Sultana, M.; Jung, S.K. Deep neural network concepts for background subtraction: A systematic review and comparative evaluation. *Neural Netw.* **2019**, *117*, 8–66. [[CrossRef](#)]
32. Babaee, M.; Dinh, D.T.; Rigoll, G. A deep convolutional neural network for video sequence background subtraction. *Pattern Recognit.* **2018**, *76*, 635–649. [[CrossRef](#)]
33. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
34. Freund, Y.; Schapire, R. A short introduction to boosting. *J. ISAI* **1999**, *14*, 771–780.
35. Dong, P.; Wang, W. Better region proposals for pedestrian detection with R-CNN. In Proceedings of the IEEE Visual Communications and Image Processing, Chengdu, China, 27–30 Nov 2016; pp. 1–4.
36. Zitnick, C.L.; Dollar, P. Edge Boxes: Locating Object Proposals from Edges. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 391–405.
37. Uijlings, J.R.R.; Van De Sande, K.E.A.; Gevers, T.; Smeulders, A.W.M. Selective search for object recognition. *Int. J. Comput. Vis.* **2013**, *104*, 154–171. [[CrossRef](#)]
38. Pan, S.; Yang, Q. A Survey on Transfer Learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [[CrossRef](#)]
39. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
40. ImageNet. Available online: <http://www.image-net.org> (accessed on 20 February 2019).
41. Zhu, X.; Goldberg, A. *Introduction to Semi-Supervised Learning*. Synthesis lectures on Artificial Intelligence and Machine Learning; Morgan & Claypool: San Rafael, California, USA, 2009; pp. 1–130.
42. Krizhevsky, A.; Sutskever, I.; Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the International Conference on Neural Information Processing Systems, Lake Tahoe, Nevada, USA, 3–6 December 2012; pp. 1097–1105.
43. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
44. Mathworks. Available online: <https://www.mathworks.com/help/vision/examples/object-detection-using-faster-r-cnn-deep-learning.html> (accessed on 20 February 2019).