PanduNet: Pedestrian detection Accurate Network with Deformable Units

Shardul Shailendra Parab
Department of Computer Science
UCLA
UID - 205525006
shardulparab@g.ucla.edu

Abirami Anbumani
Department of Computer Science
UCLA
UID: 005526158
ad14abirami@g.ucla.edu

Aishwarya Dev
Department of Computer Science
UCLA
UID - 305515097
aishwaryadev30@g.ucla.edu

Harini Suresh
Department of Computer Science
UCLA
UID - 505712718
sharini16@cs.ucla.edu

Harshita Khandelwal
Department of Computer Science
UCLA
UID - 905526234
harshitaskh@g.ucla.edu

Abstract

Object detection is extensively performed by Convolutional Neural networks(CNN) and their variations. MobileNet is one lightweight deep learning-based single shot detector with embedded vision applications. CNNs are inherently limited to model geometric transformations due to the fixed geometric structures in their building modules. Deformable models strengthen this transformation capacity by embedding object information into the system. In this paper, we describe our work titled 'PanduNet,' a Deformable Convolutions incorporated MobileNet with application in Pedestrian Detection. We have compared and summarized the results of traditional MobileNet and PanduNet for the same. We have observed that our Pandunet model does perform much better than the traditional Mobilenet for a multitude of parameters such as Mean Average Precision, different object sizes as well as differently oriented images. Furthermore, we have summarized our work on pedestrian tracking using PanduNet.

1 Introduction

Pedestrian Detection forms a major part of Surveillance, Traffic Safety, and Autonomous Vehicle functioning. Traditionally Pedestrian Detection techniques(Benenson et al. [2014]) involved using Histogram-Of-Gradients(HOG), Support Vector Machines(SVM), Deformable part models, and Computer Vision techniques. Recently, deep learning techniques with an emphasis on Convolution Neural Networks(CNN) are gaining popularity for object detection. SSD is an architecture modification to CNN which enables easier training and integration with the system.

Single Shot Detector(SSD) is a deep neural network architecture with the capability to detect multiple objects in an image frame using discretized output space of bounding boxes(Liu et al. [2016]). The ubiquity of SSD is attributed to its easy-to-train nature and ability to integrate with object detection systems. MobileNet is one such backbone architecture of SSD designed for mobile and embedded vision applications (Howard et al. [2017]). SSD backbone architecture is modified according to its utility; VGG16(Liu et al. [2016]), AlexNet(Tomè et al. [2016]), ResNet(CENGİZ et al.), SqueezeNet, MobileNet, YOLO V3 (Menon et al. [2021])are popularly mentioned in literature for Object Detection. In this research, we are using MobileNet for Pedestrian detection since they are lightweight and easy to deploy on the edge. MobileNets are lightweight deep learning architecture that leverages the concept of depth-wise separable convolutions to work with mobile applications. They work by finding the tradeoff between accuracy and latency using global hyper-parameters.

Deformable models bring the ability to detect objects in image frames with objects changing position and shape with time. Although initially developed for application in Computer Graphics(Terzopoulos and Fleischer [1988]), deformable models are currently used in Medical Image Analysis(Weese et al. [2001]), Face Recognition(Yuille et al. [1992]), Object Detection(Felzenszwalb et al. [2010]) and Segmentation(Huang et al. [2005]). Pedestrian Detection includes dynamic frames where the location and posture of pedestrians are constantly changing. Therefore in this paper, we have described our work on PanduNet, a Pedestrian detection Accurate Network with Deformable Units. It is an SSD Architecture with MobileNet as its backbone architecture and Deformable Convolution Unit incorporated in succeeding layers. We have also described our work on Real-Time Pedestrian Tracking, a subset of Multi-Object Tracking, using PanduNet.

This paper summarizes the following major contributions:

- PanduNet, formed by fusing deformable convolution with MobileNet back-boned Single-Shot Detector
- Pedestrian Detection using PanduNet
- Real-Time Pedestrian Tracking with PanduNet.

We have organized the paper into multiple sections as follows: Proposed Model, Experimental Setup, Results, Conclusion, and Future Work.

2 Proposed Model

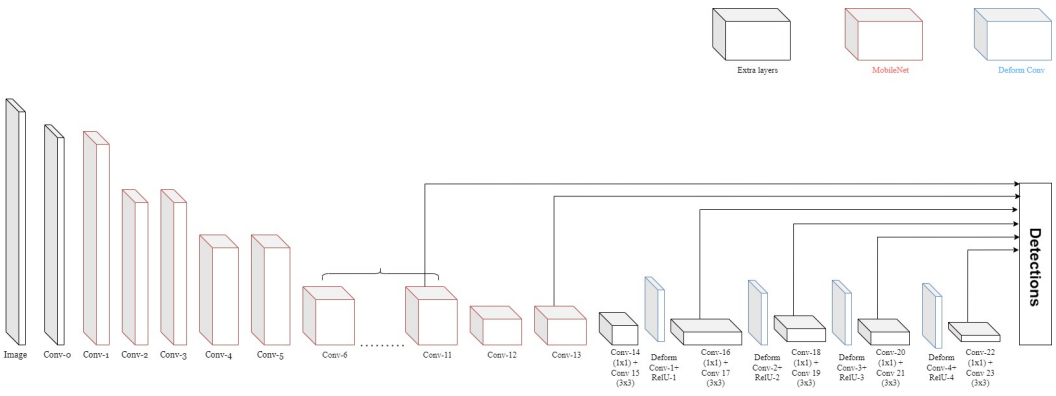

Figure 1: Proposed Architecture of the Model

Pedestrian Detection is unlike tradition object detection for the following reasons: Distance, Orientation, Shape, Pose, and size of the pedestrian affects detection. Population density is also a distinguishing factor. In order to work amidst such difficulties, we have built the model PanduNet, which is an integration of Single Shot Detector and Deformable Units.

PanduNet has the following parts:
1. Single Shot Detector with MobileNet backbone
2. Deformable Convolution Unit

Single Shot Detector
SSD consists of a base network followed by auxiliary structure which allows single shot detection(Liu et al. [2016]). The base network is a feed forward neural network with image classification capabilities. The base network we have chosen is a MobileNet. The auxiliary structure allows for multi-scale feature map based-detection, convolution predictors based-detection, default boxes with multiple aspect ratios for prediction.

The SSD architecture used two error functions as metric for training: Localization and Confidence Loss.

- Total loss is a sum of localization and confidence loss.

$$L(x, c, y, g) = \frac{1}{N} L_{conf}(x, c) + L_{loc}(x, y, g)$$

- Localization loss compares predicted output with ground truth with a smooth L1 loss.

$$L_{\text{loc}}(x, y, g) = \sum_{i \,\epsilon\, Pos} \sum_{m \,\epsilon\, \{cx,cy,w,h\}} x_{ij}^{k} smooth_{L1}(y_i^m - \hat{g}_i^m)$$

$$\hat{g}_j^{cx} = (g_j^{cx} - d_i^{cx}))/d_i^w$$
$$\hat{g}_j^{cy} = (g_j^{cy} - d_i^{cy}))/d_i^h$$
$$\hat{g}_j^{w} = log\left(\frac{g_j^w}{d_i^w}\right)$$
$$\hat{g}_j^{h} = log\left(\frac{g_j^h}{d_i^h}\right)$$

- Confidence Loss which is computed for multiple classes during prediction. It is a softmax function given as follows:

$$L_{conf}(x, c) = -\sum_{i \,\epsilon\, Pos}^{N} x_{ij}^{p} log(\hat{c}_i^p) - \sum_{i \,\epsilon\, Neg}^{N} log(\hat{c}_i^0)$$

where,
L_{loc} - Localization Error
y - Predicted Output
g - Ground Truth
cx, cy - Centre Point Coordinates
w - Width
h - Height
c - class confidence
p - category
$x^p_{ij} = \{1, 0\}$, matches i-th default element to j-th ground truth element of p

2.1 MobileNet Backbone

MobileNet (Howard et al. [2017]) is a lightweight architecture which incorporates depth-wise separable convolution units to enable easy training and deployment.
The following are the major components of a MobileNet:
1. Depthwise Seperable Convolution

2. Width Multiplier
3. Resolution Multiplier

Depthwise Seperable Convolution is based on the concept of discretizing standard convolution into depthwise and pointwise convolution.

Standard Convolution Output is $G_{k,l,n} = \sum_{i,j,m} K_{i,j,m,n} F_{k+i1,l+j1,m}$
$Where,$
$K - ConvolutionalKernel$
$F, G - FeatureMap$

The computational cost for depthwise seperable convolution is $D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + M \cdot N \cdot D_F \cdot D_F$

By discretizing a layer into two layers, the computation is reduced by $1/N + 1/D^2_K$ **Width Multiplier** Certain applications require faster and much simpler version than the low latency Mobile Network. This is obtained by using width multiplier α.

When introduced a width multiplier the input and output channels are multiplied by a factor of α.

The computational cost after introducing width multiplier is $D_K D_K \alpha M D_F D_F + \alpha M \alpha N D_F D_F$.

Resolution Multiplier is also introduced to reduce the computational cost,

$$D_K D_K M D_F D_F + M N D_F D_F$$

where the resolution multiplier is ρ.

2.2 Auxiliary structure

The auxiliary structure in a SSD consists of Convolution layers grouped to perform detection based on:
1. Multi-scale Feature Map
2. Convolution Predictors
3. Default Boxes.

We have incorporated Deformable Units in addition to Convolution layers to build PanduNet.

2.2.1 Deformable Units

Deformable unit adapts to geometric variation of objects. Sometimes, this units adaptively fits to irrelevant image content. To overcome this, we have incorporated Comprehensive integration of deformable unit with convolution layer and modulation mechanism for deformable modelling (Zhu et al. [2019])

Modulated Deformable Convolution is

$$y(p) = \sum_{k=1}^{K} w_k x(p + p_k + \Delta p_k) \Delta m_k$$

where,
$\Delta m_k - Modulation\ Scalar$
$\Delta p_k - Learnable\ Offset$
$x(p + p_k + \Delta p_k) - Bilinear\ Interpolation$

2.2.2 ReLu and Bottle Neck Convolution Block

Deforamble Unit is followed by ReLu to facilitate non-linear transformation. A Bottle neck convolution block is included to ensure dimensionality reduction while maintaining consistency.

2.3 Module for Pedestrian Tracking

Multi Object Tracking(MOT) is an expensive activity if one uses the deep vector features in order to track objects in a video frame. Instead, we aim to make the system fast and inexpensive by using a tracking system which depends on using only the bounding box results, scores and distances between the objects in order to tracking pedestrians in the live video. Additionally, the system also works as a highly robust frame aggregator in order to improve the MAP of the model to a great extent. Likewise, herein we hope that the integrate the new model with such a tracker which aid in being efficient both speed and accuracy wise and also be lightweight at the same time.

3 Dataset

We performed an extensive research to understand diverse range of datasets to create an all inclusive class balanced training dataset. We chose an ensemble of datasets to prevent bias in learning and test model under diverse conditions such as darkness, crowd, etcetera.

This ensemble include P-DESTRE, Penn-Fudan, CrowdHuman, Caltech, and Pascal VOC. Please find a summary of the datasets in Table 1.

Dataset	**P-DESTRE**	**Penn-Fudan**	**CrowdHuman**	**Caltech**
Environment	Outdoor	Outdoor	Indoor, Outdoor	Outdoor
Number of Images/Frames	90 Videos, 4 - 2476	170	15000	42782
Number of People	14.2M	345	339565	13674
Density(People/ Image)	63.96(Bounding box)	2.03	22.64	0.32
Occlusion?	High	Medium	High	Low
Blur	Yes	No	No	No
Tracking	Yes, Re-identification	No	No	Yes
Format	Video	Still	Still	Video

Table 1: Dataset Comparison

We have trained PanduNet using Pascal VOC Dataset with human in image frames. In future, we aim to train on the ensemble of dataset to make the model more efficient and inclusive.

4 Observation and Results

4.1 Experimental Results

The experiments are based on a comparative study between a mobilenet SSD model (without any deformable convolutional units) and the PanduNet model which promises to produce much better results if allowed to train for the entire run i.e. 200 epochs. As explained previously, we do add and alter the base Mobilnet SSD model by adding deformable convolution units in all the extra layers after the mobilenet model. Currently, we set out to train both the models using the transfer learning approach by freezing the Mobilnet[1] Architecture trained on Imagenet dataset[2]. In ideal conditions, the expected training schedule should include using a batch size of 32 for 200 epochs using a cosine scheduler with an initial learning rate of 0.01. However, due to the current lack of resources we have to have done our comparative study on a highly pessimistic training plan, training for 23 epochs on the Pascal VOC dataset for both the base Mobilenet SSD and Pandunet using 1500 data points for training and 300 images for testing with a batch size of 4(cannot exceed the batch size 4 due to memory constraints). The training is completed on a 4GB 1050Ti NVIDIA GPU and the

codebase is majorly written using the Pytorch framework in Python.

As described in the Mobilenet paper, we will use the essential data augmentation steps in order to avoid any gaps in the training process. The paper states the following steps:
– Use the entire original input image.
– Sample a patch so that the minimum jaccard overlap with the objects is 0.1, 0.3, 0.5, 0.7, or 0.9.
– Randomly sample a patch.

We will do a comparative study on the following factors:
a) Accuracy comparison using MAP on the PASCAL VOC dataset.
b) Accuracy comparison with pedestrians at different size scales i.e. small, medium and large.
c) Visual Study for objects with different orientations.
d) Visual study of handling occlusions.

4.1.1 Mean Average Precision

Precision and recall are single-value measurements based on the system's entire list of documents returned. It is desirable to consider the order in which the returned documents are presented when using systems that return a rated sequence of documents. One can create a precision-recall curve by computing precision and recall at each location in the ranking sequence of documents and graphing precision $p(r)$ as a function of recall r. The average value of $p(r)$ for the range of $r = 0$ to $r = 1$ is computed using average precision:

$$AveP = \int_0^1 p(r)dr$$

Mean average precision (MAP) for a set of queries is the mean of the average precision scores for each query.

$$MAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q}$$

4.1.2 Results on PASCAL VOC Dataset

As stated previously, we are using a subset of the Pascal VOC dataset using 1500 data points and 300 images. The metric used for the following is the Mean Average Precision. In the current case after 23 epochs the MAP score of the base mobilenet SSD model is 32.39 whereas for our Pandunet we get an MAP of 43.59 at only 23 epochs which is a huge improvement. This also shows that our model has a tendency to converge much earlier and in a much more accurate manner (as seen in Figure 2 and 3). Lastly, we predict that if we had more resources and time, the model would have definitely shown a vast improvement over the previously existing MobileNet SSD.

Figure 2: Results for MobileNet + SSD Model

Figure 3: Results for MobileNet + SSD + Deformable Model

4.1.3 Results on Accuracy comparison with pedestrians at different size scales i.e. small, medium and large.

The MAP score is an indication of the entire dataset but does not give an idea about the minute details. So we have also conducted a study on pedestrians of different sizes i.e. small, medium, large, etc. It could be a case that the model could become biased to only images of a particular size and it can definitely help in dissecting the model for further improvements. For this case we use the COCO dataset and compare small, medium and large. These Average Precision Scales are strictly predefined as objects smaller than 32x32 to greater than 96x96 pixels. The results of the same are as follows: small - mobilenetSSD is 25.32 and Pandunet is 33.42. For medium scaled images - mobilenetSSD is 28.66 and Pandunet is 33.96. However, for large scaled humans in the dataset, mobilenetSSD scores 27.33 and Pandunet scores 27.12. So in the majority of cases, Pandunet outscores mobilenetSSD and can be generalized for pedestrian detection, which can be concluded from Figure 4 and Figure 5. Additionally, it is expected that for pedestrians the small and medium sized humans would be the case in general and large sized cases would be rare. Although in the future, it needs to be seen as to why large objects are not giving optimal performance with Pandunet.

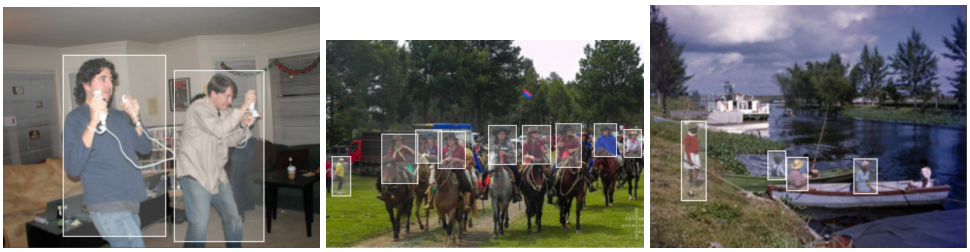

Figure 4: Results for MobileNet + SSD Model

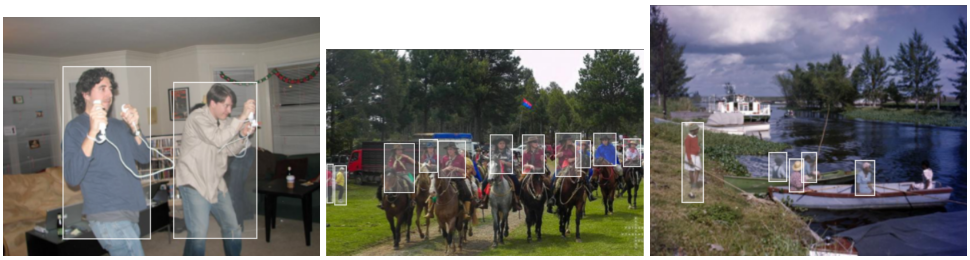

Figure 5: Results for MobileNet + SSD + Deformable Model

4.1.4 Visual Study for handling objects with different orientations:

As per the paper on Deformable Convolutions, it is expected that the detector using Deform Convolutions should work with objects at different orientations and give consistent results for the same. To

test this hypothesis, we run our model on images with different orientations and test how well the model performs. Based on testing a few detectors such as Yolov1, mobilenet-SSD It is evident that object detectors are in general not actually trained for this scenario. However, upon evaluating the performance of PanduNet on a very small dataset prepared by the team by rotating images, we have observed a positive performance, as determined in Figures 6 and 7. This does present optimism that the model can give really good results upon complete training.

Figure 6: Results for MobileNet + SSD Model

Figure 7: Results for MobileNet + SSD + Deformable Model

4.1.5 Visual study of handling occlusions:

Detection of occluded pedestrians is a big issue that needs to be solved for consistent detection as well as tracking. The salient features of deformable convolutions does provide a reason to believe that PanduNet can handle occlusions too to a considerable extent. We present a few useful cases where PaduNet has successfully managed to provide positive results of occluded/hidden pedestrians. This is also a positive outcome wherein upon complete training, Pandunet can also help in providing consistent trackers and help develop consistent and robust detectors in the long term. Figures 8 and 9 paint a positive picture for the same.

The final results obtained can be complied into a table as follows (Refer Table 2):

Field	MobileNet SSD	PanduNet
Pascal VOC (300 Subset)	32.39	43.59
Small Targets	25.52	33.42
Medium Targets	28.66	33.96
Large Targets	27.33	27.12

Table 2: Comparison of results on different data.

Figure 8: Results for MobileNet + SSD Model

Figure 9: Results for MobileNet + SSD + Deformable Model

4.1.6 Module for Pedestrian Tracking

PanduNet is built explicitly to build the tracker on a consistent basis with bounding boxes, at every specific time interval. Whereas, for the MobileNet SSD model, we predict that the bounding boxes are not framed consistently and hence the model is not accurately predicting the tracker unlike the PanduNet model proposed by us. (Refer Figure 10 and Figure 11)

5 Conclusion

We designed PanduNet which is a Pedestrian detection Accurate Network with Deformable Units. This network is a deformed single shot detector with its backbone architecture as MobileNet and auxiliary network as deformable convolution units. We tested and compared PanduNet and Mobilenet-SSD on PASCAL VOC dataset and COCO dataset for MAP and image sizes respectively and clearly observed that our Pandunet model does perform relatively much better. The results also showcase the power of deformable convolutions and can definitely enhance the performance of detectors to a great extent.

6 Future Work

We hope to extend our work as follows:

Figure 10: Results for MobileNet + SSD Model

Figure 11: Results for PanduNet

- Train PanduNet a diverse set of Pedestrian Detection Datasets to better understand what the architecture is learning
- Improve the Auxiliary layer of the architecture and test the model
- Change the backbone architecture from MobileNet to VGG16, AlexNet, ResNet, and SqueezeNet to compare their performances
- Train and Test PanduNet for other Object Detection applications

References

Junfeng Bai, Zongqing Lu, and Qingmin Liao. Single shot relation detector for pedestrian detection. In *Eleventh International Conference on Digital Image Processing (ICDIP 2019)*, volume 11179, page 1117929. International Society for Optics and Photonics, 2019.

Hatem Belhassen, Heng Zhang, Virginie Fresse, and El-Bay Bourennane. Improving video object detection by seq-bbox matching. In *VISIGRAPP (5: VISAPP)*, pages 226–233, 2019.

Rodrigo Benenson, Mohamed Omran, Jan Hosang, and Bernt Schiele. Ten years of pedestrian detection, what have we learned? In *European Conference on Computer Vision*, pages 613–627. Springer, 2014.

Enes CENGİZ, Cemal YILMAZ, and Hamdi KAHRAMAN. Classification of human and vehicles with the deep learning based on transfer learning method. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 9(3):215–225.

Enhui Chai and Min Zhi. Rapid pedestrian detection algorithm based on deformable part model. In *Ninth International Conference on Digital Image Processing (ICDIP 2017)*, volume 10420, page 104200Q. International Society for Optics and Photonics, 2017.

Hyunggi Cho, Paul E Rybski, Aharon Bar-Hillel, and Wende Zhang. Real-time pedestrian detection with deformable part models. In *2012 IEEE Intelligent Vehicles Symposium*, pages 1035–1042. IEEE, 2012.

Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017.

Pedro F Felzenszwalb, Ross B Girshick, and David McAllester. Cascade object detection with deformable part models. In *2010 IEEE Computer society conference on computer vision and pattern recognition*, pages 2241–2248. Ieee, 2010.

Ujwalla Gawande, Kamal Hajari, and Yogesh Golhar. Pedestrian detection and tracking in video surveillance system: issues, comprehensive review, and challenges. *Recent Trends in Computational Intelligence*, 2020.

Fen He, Paria Karami Olia, Rozita Jamili Oskouei, Morteza Hosseini, Zhihao Peng, and Touraj BaniRostam. Applications of deep learning techniques for pedestrian detection in smart environments: A comprehensive study. *Journal of Advanced Transportation*, 2021, 2021.

Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

Xiaolei Huang, Zhen Qian, Rui Huang, and Dimitris Metaxas. Deformable-model based textured object segmentation. In *International workshop on energy minimization methods in computer vision and pattern recognition*, pages 119–135. Springer, 2005.

Matthieu Lin, Chuming Li, Xingyuan Bu, Ming Sun, Chen Lin, Junjie Yan, Wanli Ouyang, and Zhidong Deng. Detr for crowd pedestrian detection. *arXiv preprint arXiv:2012.06785*, 2020.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

Aiswarya Menon, Bini Omman, and S Asha. Pedestrian counting using yolo v3. In *2021 International Conference on Innovative Trends in Information Technology (ICITIIT)*, pages 1–9. IEEE, 2021.

Alessandro Preziosi, Antonio Prioletti, and Luca Castangia. Faster pedestrian recognition using deformable part models. *International Journal of Computer and Information Engineering*, 10(10): 1741–1750, 2016.

Tso-Liang Teng and Trung-Kien Le. Development and validation of a pedestrian deformable finite element model. *Journal of mechanical science and technology*, 23(8):2268–2276, 2009.

Demetri Terzopoulos and Kurt Fleischer. Deformable models. *The visual computer*, 4(6):306–331, 1988.

Denis Tomè, Federico Monti, Luca Baroffio, Luca Bondi, Marco Tagliasacchi, and Stefano Tubaro. Deep convolutional neural networks for pedestrian detection. *Signal processing: image communication*, 47:482–489, 2016.

Jürgen Weese, Michael Kaus, Christian Lorenz, Steven Lobregt, Roel Truyen, and Vladimir Pekar. Shape constrained deformable models for 3d medical image segmentation. In *Biennial International Conference on Information Processing in Medical Imaging*, pages 380–387. Springer, 2001.

Han Xie, Wenqi Zheng, and Hyunchul Shin. Occluded pedestrian detection techniques by deformable attention-guided network (dagn). *Applied Sciences*, 11(13):6025, 2021.

Alan L Yuille, Peter W Hallinan, and David S Cohen. Feature extraction from faces using deformable templates. *International journal of computer vision*, 8(2):99–111, 1992.

Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9308–9316, 2019.