Velastin, S.A. y Gómez-Lira, D.A. (2017). People Detection and Pose Classification Inside a Moving Train Using Computer Vision. In *Advances in Visual Informatics. Lecture Notes in Computer Science*, 10645, pp. 319-330.

DOI: https://doi.org/10.1007/978-3-319-70010-6_30

People Detection and Pose Classification Inside a Moving Train Using Computer Vision

Sergio A. Velastin[1,2(✉)] and Diego A. Gómez-Lira[3]

[1] Department of Computer Science, Universidad Carlos III de Madrid, Colmenarejo 28270, Madrid, Spain
sergio.velastin@ieee.org
[2] Queen Mary University of London, London, UK
[3] Department of Informatics Engineering, Universidad de Santiago de Chile, Santiago, Chile
diegog.asd@gmail.com

Abstract. The use of surveillance video cameras in public transport is increasingly regarded as a solution to control vandalism and emergency situations. The widespread use of cameras brings in the problem of managing high volumes of data, resulting in pressure on people and resources. We illustrate a possible step to automate the monitoring task in the context of a moving train (where popular background removal algorithms will struggle with rapidly changing illumination). We looked at the detection of people in three possible postures: Sat down (on a train seat), Standing and Sitting (half way between sat down and standing). We then use the popular Histogram of Oriented Gradients (HOG) descriptor to train Support Vector Machines to detect people in any of the predefined postures. As a case study, we use the public BOSS dataset. We show different ways of training and combining the classifiers obtaining a sensitivity performance improvement of about 12% when using a combination of three SVM classifiers instead of a global (all classes) classifier, at the expense of an increase of 6% in false positive rate. We believe this is the first set of public results on people detection using the BOSS dataset so that future researchers can use our results as a baseline to improve upon.

Keywords: People detection · Posture classification · People monitoring · On-board surveillance · Machine learning

1 Introduction

The use of surveillance cameras for the prevention and management of criminal incidents is becoming increasingly common. Santiago de Chile saw in 2010 an increase of 78% in the number of cameras (313 extra cameras in 24 city districts) with an investment of over 800 thousand US dollars [1]. Furthermore, in 2013 Metro Santiago announced that it will install surveillance cameras in the more than 185 new trains to become operational in the next few years. Elsewhere, the UK is one of the world leaders in this field, [2] cites a report from Big Brother Watch claiming that there are about 51,000 police-run cameras in urban areas, with an investment of 807 million euros in the last four years and the country estimated to have 20% of the security

cameras in the world. Nevertheless, it has been reported that only one crime is solved for every 1,000 cameras [3] highlighting the need for automatic means of detecting unusual situations and where computer vision can assist. It is assumed here that the reader is reasonably familiar with computer vision and image processing techniques. In the context of this work, we focus on machine learning techniques, specifically Support Vector Machines (SVMs), e.g. see [4, 5] and pedestrian detection using Histograms of Oriented Gradients [6]. Learning machines are used to recognize patterns typically using labeled examples [7]. Popular learning machines in computer vision include SVMs [8], Adaboost [9] and neural networks (including the increasingly successful "deep" learners [10–12]). There are surprisingly little reports on the use of BOSS [13]. Truong Cong et al. [14, 15] report a foreground estimation and person re-identification approach, but it is not clear how it can deal with stationary people, nor they consider different postures. More recently Coniglio et al. [16] present an interesting approach based on shape priors, HOG and multiple SVMs but it is not clear how they deal with different poses.

In this work, we look at the classification performance of the HOG descriptor combined with a binary SVM. In our case, the observed people are contained within a normalized window of 128×256 pixels with a block size of 16 pixels, with an overlap of 8 pixels, cells of 8×8 pixels, 4 cells/block and 9 bins. Therefore, the HOG descriptor is of size 16740 (floating point numbers). The overall idea is to collect sufficient samples of people (at the different poses that need to be classified) and of *not people* (negative samples), compute the corresponding descriptors and train a classifier to separate such sample populations. In the sections that follow, we will first describe the dataset and the procedure to obtain such samples and then the results of the classification process.

2 The Dataset

For this work, we used the public BOSS dataset as it contains a series of semi-realistic videos of people acting out incidents (such as thefts, fights, fainting, etc.) as well as normal behavior inside a moving train (thus with sometimes rapidly changing illumination), using 9–10 cameras. The dataset has a sixteen video sequences (the lan-guage refers to the audio that was also recorded), including:

1. Cell phone Spanish: struggle between two people and theft of a cell phone.
2. Checkout French: appearance of three people, in which a fight between two of them occurs.
3. Disease: Person fainting.
4. Disease public: Person fainting, plus six people helping.
5. Faces (4 sequences): different people walking along the train corridor.
6. Harass French: harassment of a passenger by another.
7. Harass Spanish: harassment of a passenger by another.
8. Harass French 2: harassment of a passenger by another plus 5 witnesses.
9. Newspaper French: fight between two passengers over a newspaper with 4 witnesses.

10. Newspaper Spanish: fight between two passengers over a newspaper with 4 witnesses.
11. No event (2 sequences): Between seven to ten people talking or greeting each other.
12. Panic: Eleven people fleeing the train.

Ultimately, it will be useful to investigate human action/interaction recognition algorithms that could identify the above situations. However, there are not enough examples on this video set to train such systems and in any case, before actions/interactions could be detected a system would typically need first to detect/track each individual in the images and their postures. As this is still a challenging problem (especially when dealing with multiple postures and rapid illumination changes), that is what we concentrate on in this paper. For our experiments, we used data from camera 1, an example of which is shown in Fig. 1.

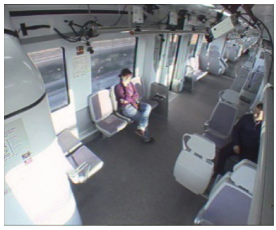

Fig. 1. View from camera 1 (BOSS dataset)

The BOSS dataset only provides annotations at the level of actions/interaction and not of the position and posture of each person, so we had to create such ground truth (this is available upon request from the authors). We used the VIPER-GT [17] annotation tool, using three attributes for each person every eight frame (to save some effort): ID (a unique identifier assigned to a person when he/she appears for the first time in the scene), Body (a rectangular bounding box) and Status (the posture class coded as 0: Sat down, 1: Sitting (half-way between Sat down and Standing), and 2: Standing).

2.1 Ground-Truthing

The ground-truthing process consists of five sequential stages:

Video Labeling.
This process consists on using VIPER-GT to manually localize and label each person in each frame of the video (this is therefore the most time-consuming part) as illustrated in Fig. 2a.

Pre-processing.
Once the frames have been labeled, the images for each labeled pedestrian are extracted to be used later as positive training samples for the classifiers. This is illustrated in

(a) (b)

Fig. 2. (a) Labeling each person in the video (ID, bounding box and status), (b) extracted pedestrians

Fig. 2b. It should be noted that in fact, we extract an additional border of 10% of the annotated bounding boxes, following the finding in [6] that the inclusion of some background improves the performance of the classifier. These are referred as *expanded* bounding boxes/images.

Under the hypothesis that we want to avoid too much occlusion in the positive training samples, a further check is done on bounding boxes in each frame. For a given *expanded* bounding box, its intersection I_{eu} with other non-expanded bounding box in the same frame is calculated by:

$$I_{eu} = \frac{A_e \cap A_u}{A_e \cup A_u} \tag{1}$$

where A_e is the area of the expanded box and A_u the area of an unexpanded box. If the intersection is *greater* than a given threshold (τ_{ov}) then the candidate expanded sample is *discarded*. An illustration of intersection is shown in Fig. 3 To evaluate the effect of this process of positive samples selection, we have conducted tests for values of τ_{ov} = 1.0, 0.5 and 0.2 yielding 16323, 13852 and 6698 positive samples respectively. Note that a value of 1.0 means that all samples are accepted even if they are fully occluded.

Fig. 3. Intersection of bounding boxes

Extraction of Positive Images.
This consists on using the ground truth annotation and the intersection rule to extract positive samples (including the additional border). An illustrative example is shown in Fig. 4.

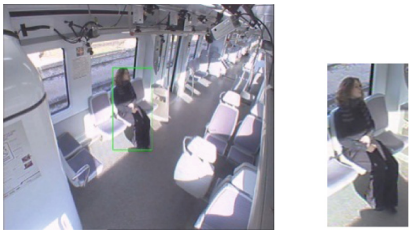

Fig. 4. An image and the corresponding extracted positive sample (in this case of status "Sat down")

Image Normalization.
A restriction with most learning machines (including SVM) is that the feature vectors for all samples (positives and negatives) need to be of the same dimension. An additional restriction in our case is that the OpenCV implementation of HOG needs the size (horizontal and vertical) of the images to be a multiple of 8. As pedestrian images in the original annotations vary in size, we have used OpenCV's *resize* function to resize all positive samples obtained after stage 3 above, to a size of 128 $\times$ 256 (these are close to the mean sizes of the annotated images).

Extraction of Negative Examples.
Negative samples are image regions (normalized as explained above) that contain no people. Given the ground truth, we know where people are and so what can be done is to sample the video images randomly making sure that such samples do not overlap regions containing people. In this way, a very large number of negative samples (compared to the number of positives) can be obtained. To reduce the size of such population, we use the following rules (in each case the resulting negative samples are then size normalized to 128 $\times$ 256):

- For each frame without people, we obtain five random (location and size) negative images (the literature tells us that a ratio of five negatives for one positive is popular).
- In frames with people, for each person we get a random sample of the same size and checking that it does not overlap more than a given fraction (as above) with any annotated person.

In this way, a total of 23990 negative samples were obtained. That population of negative samples is maintained constant for all the experiments so as not introduce a random element. This will also allow a better comparison of algorithms should other researchers wish to use the same data (available upon request from the authors).

2.2 Sample Groups and Sizes

To evaluate the performance of various machines, different groups of positives, depending on the position of people and the intersection with others were used:

By Posture.

We separated the positive samples into four groups. The three first groups correspond to samples in each of the posture categories (Sat down, Sitting, Standing) while the last group contains all samples irrespective of their posture and we called this **Full**.

By Intersection.

These correspond to the groups obtained by varying the maximum allowed overlap as explained earlier and called "**1.0**", "**0.5**" and "**0.2**".

The above groupings resulted in the following numbers of positive samples (Table 1):

Table 1. Positive samples groupings

1.0 (100%)	0.5 (50%)	0.2 (20%)
Sat down: 11218	Sat down: 9232	Sat down: 3428
Sitting: 1198	Sitting: 1079	Sitting: 640
Standing: 3907	Standing: 3541	Standing: 2630
Full: 16323	Full: 13852	Full: 6698

3 Experimental Classification Results

The descriptors for each of the negative and samples can be calculated using OpenCV's *compute()* method in its *HOGDescriptor* class. The different groupings outlined above form the basis for different experiments. In each case, we use OpenCV's SVM implementation to train classifiers. We use 10-fold cross validation meaning that in each group we take 10% of the samples as a testing set and the remaining 90% as a training set, computing results (mean, variances) for 10 possible 10%:90% partitions. We compute *sensitivity (S)* and *false positive rate (FPR)*:

$$S = \frac{TP}{TP + FN}, \quad FPR = \frac{FP}{TP + FP} \tag{2}$$

where *TP* is the number of true positives, *FN* the number of false negatives and *FP* the number of false positives.

3.1 Global vs. Specific SVM

The first test is to evaluate the performance of a classifier that considers all postures and compare it with classifiers trained on specific postures to see if such specialization is useful for the purposes of pedestrian detection (independently of postures). The aggregated cross-validation results for sensitivity and FPR are shown in Table 2.

These indicate that in all cases, the global classifier has a poorer performance than the more specialized classifiers. The results for the other two intersections are given in Tables 3 and 4 that confirm this finding and show that results tend to improve with less occluded training samples (but this effect might be due to the smaller variability in the less occluded samples, recalling that these resulted from discarding positive samples).

Table 2. Mean sensitivity and FPR for intersection 1.0 (ERROR is standard deviation)

	Sensitivity	ERROR	FPR	ERROR
Global	0.790	0.053	0.253	0.109
Sat down	0.817	**0.037**	0.128	0.052
Sitting	0.830	0.043	**0.056**	**0.017**
Standing	**0.850**	0.043	0.132	0.026

Table 3. Mean sensitivity and FPR for intersection 0.5

	Sensitivity	ERROR	FPR	ERROR
Global	0.810	0.035	0.205	0.095
Sat down	0.857	0.057	0.124	0.066
Sitting	0.868	0.037	**0.045**	**0.009**
Standing	**0.889**	**0.020**	0.097	0.017

Table 4. Mean sensitivity and FPR for intersection 0.2

	Sensitivity	ERROR	FPR	ERROR
Global	0.860	0.031	0.158	0.070
Sat down	**0.941**	**0.021**	0.058	0.022
Sitting	0.877	0.045	**0.039**	0.014
Standing	0.907	0.022	0.073	**0.011**

Overall, the more specific classifiers show an improvement of between 1–8% in sensitivity and 8-20% in FPR.

3.2 Use of Cross-Negatives

By "cross" negatives we mean people samples from a different class that are used as negative training samples (e.g. using "Sat down" positive samples as negative samples to train a "Standing" classifier). We have experimented with adding different proportions of such samples (10%, 20% and 50% taken equally from the remaining two classes), obtaining the results shown in Tables 5, 6 and 7. In none of these cases the addition of this type of negative samples improved the performance of the individual of these classifiers.

Table 5. Mean sensitivity and FPR for "Sat Down" classifier, using cross negatives

	Sensitivity	ERROR	FPR	ERROR
Sat down	**0.941**	**0.021**	**0.058**	**0.022**
Sat down + 10%	0.914	0.032	0.066	0.023
Sat down + 25%	0.880	0.028	0.077	0.031
Sat down + 50%	0.859	0.031	0.084	0.036

Table 6. Mean sensitivity and FPR for "Sitting" classifier, using cross negatives

	Sensitivity	ERROR	FPR	ERROR
Sitting	**0.877**	**0.045**	**0.039**	**0.014**
Sitting + 10%	0.777	0.066	0.067	0.016
Sitting + 25%	0.719	0.108	0.082	**0.014**
Sitting + 50%	0.730	0.058	0.106	0.019

Table 7. Mean sensitivity and FPR for "Standing" classifier, using cross negatives

	Sensitivity	ERROR	FPR	ERROR
Standing	**0.907**	**0.022**	**0.073**	**0.011**
Standing + 10%	0.886	0.032	0.103	0.022
Standing + 25%	0.847	0.062	0.111	0.026
Standing + 50%	0.829	0.034	0.122	0.020

3.3 General vs. Combined Classifier

In this experiment, we investigate if the better performance we have seen in the individual classifiers could be maintained by combining them as an alternative to the global classifier. The steps in this experiment are:

- Separate training: Each classifier is independently trained as reported earlier (using an intersection of 0.2). The negative samples set is the same for each classifier.
- Separate testing: For each test image, each classifier is evaluated.
- Combination by voting: if at least one classifier indicates the presence of a person, then a person is deemed to have been detected. If no classifier detects a person, then no person is detected.

The results of this type of combination are shown in Table 8. There is an improvement in sensitivity of 11.8%, but at the expense of an increase in FPR of 6.3% (as it might be expected given the combination rule).

Table 8. Sensitivity and FPR for the global and the combined classifiers (intersection = 0.2)

	Sensitivity	ERROR	FPR	ERROR
Global	0.860	0.031	**0.158**	0.070
Combination	**0.978**	**0.007**	0.221	**0.058**

3.4 Using the Distance to the Hyperplane

In all the above experiments, we have used the default behavior of the SVM software, which returns the distance to the hyperplane. If this distance is negative the result is taken as a person detection, otherwise a no person detection. In other words, the decision boundary is at zero. It might be useful to explore if performance could be improved by changing this boundary. Figure 5 shows the distribution of the distance to

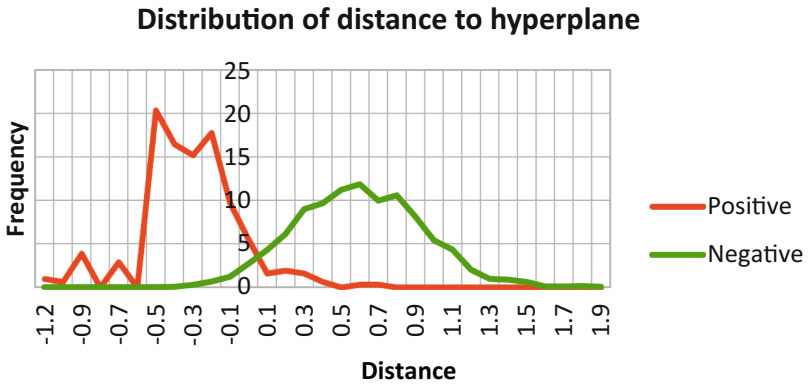

Fig. 5. Distribution of distance to hyperplane for positive and negative samples ("Sat down")

the hyperplane for positive and negative samples for the "sat down" class. Table 9 shows the results obtained when the decision boundary is −0.1, −0.2 and −0.3 compared to the original 0.0. Similarly (to save space we omit here the distance frequency graphs), Tables 10 and 11 show the results for classes "sat down" and "sitting", respectively. It is clear than in all cases there is an increase in sensitivity but at the expense of a noticeable larger increase in the FPR (effectively the operating point in a

Table 9. Sensitivity and FPR ("Sat down") for varying distances to hyperplane

	Sensitivity	ERROR	FPR	ERROR
Original	0.941	0.021	**0.058**	**0.022**
0.1	0.972	0.011	0.110	0.036
0.2	0.988	0.007	0.192	0.052
0.3	**0.999**	**0.002**	0.413	0.082

Table 10. Sensitivity and FPR ("Sitting") for varying distances to hyperplane

	Sensitivity	ERROR	FPR	ERROR
Original	0.877	0.045	**0.039**	**0.014**
0.1	0.975	0.023	0.148	0.033
0.2	0.981	**0.021**	0.237	0.046
0.3	**0.992**	0.447	0.345	0.186

Table 11. Sensitivity and FPR ("Standing")

	Sensitivity	ERROR	FPR	ERROR
Original	0.907	0.022	**0.073**	**0.011**
0.1	0.959	0.015	0.142	0.020
0.2	0.982	0.010	0.241	0.028
0.3	**0.996**	**0.003**	0.360	0.040

Receiver Operating Curve, ROC, changes in a typical manner at the top end of sensitivity). Should it be necessary to minimize FPR, a decision boundary of zero would work best.

4 Conclusions

As far as we are aware, this is the first attempt at investigating and assessing pedestrian detection using the BOSS dataset and in particular in an environment subject to rapidly changing illumination conditions because of the movement of the train. A time-consuming manual ground-truthing process has been carried out and the resulting data is made available to the research community upon request for further progress in this field. We verified, through cross-validation, that classification is relatively stable as the standard deviations are within 0.1.

We also showed that the use of specialized classifiers (trained for each posture class) produce better results, in terms of sensitivity and FPR, than a classifier trained just for pedestrian presence. We also looked at the effect of fine-tuning the positives training sets by discarding samples that presented occlusion and saw that "cleaner" samples (with a stricter intersection rule) resulted in better classification performance. We rejected the hypothesis that negative samples containing pedestrians in different postures would improve posture-specific classification. This seems to indicate that although the samples are of different postures, they are sufficiently close to the positives and far from the negatives to create confusion in the classifier. We also looked at a combined classifier used to classify people presence. The motivation here being to exploit the higher sensitivity of the specialized classifiers to solve a "global" (posture-independent) pedestrian classification problem. Although sensitivity improved noticeably, so did the FPR, indicating that we moved too far to the right on the operating curve. As the combination rule is relatively simple, there is significant scope for improvements. For example, it would be possible to use the distance output of each classifier (we also saw how using this distance as a decision boundary changed the operating point of a classifier) as an indication of confidence to be used as weighted vote (better still, the output of each SVM could be converted to a pseudo-probability [18] which is a better indication of normalized confidence than an absolute distance). We hope that we have established a good baseline using a public dataset so that researchers elsewhere can compare their results to improve performance in this field. It should be noted that here we have concentrated on the classification problem i.e. given an image extract (of the same size as used for training) what is the probability that a classifier will correctly label it as a pedestrian (what we called the "global" classification) or as a pedestrian in each posture (the specific classification). Once a classifier has been evaluated, there remains the localization problem i.e. given a complete image, where are the people and in what posture. This is traditionally solved using a sliding window to scan the image in all possible locations and at different scales (because people would have different sizes as the ones used for training). This is not a trivial problem because as the sliding window starts approaching a person, the classifier would typically start responding positively and that will generate multiple hits that have be "cleaned up" (typically using non-maximal suppression). In [18] we show how this

problem is significantly eased by first characterizing the response of a classifier in terms of probability.

Acknowledgments. The work described here was carried out as part of the OBSERVE project funded by the Fondecyt Regular Program of Conicyt (Chilean Research Council for Science and Technology) under grant no. 1140209. S.A. Velastin is grateful to funding received from the Universidad Carlos III de Madrid, the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no. 600371, el Ministerio de Economía y Competitividad (COFUND2013-51509) and Banco Santander.

References

1. La Tercera: Cámaras de seguridad en la Región Metropolitana aumentarán en un 78% (2010). http://www.latercera.com/noticia/camaras-de-seguridad-en-la-region-metropolitana-aumentaran-en-un-78. Accessed 24 June 2017
2. Evans, I.: Report: London no safer for all its CCTV cameras (2012). http://www.csmonitor.com/World/Europe/2012/0222/Report-London-no-safer-for-all-its-CCTV-cameras. Accessed 24 June 2017
3. BC News: 1,000 cameras 'solve one crime' (2009). http://news.bbc.co.uk/2/hi/8219022.stm. Accessed 24 June 2017
4. Burges, C.J.: A tutorial on support vector machines for pattern recognition. Data Min. knowl. Discov. **2**(2), 121–167 (1998)
5. Nowozin, S., Lampert, C.H.: Structured learning and prediction in computer vision. Found. Trends® Comput. Graph. Vis. **6**(3), 185–365 (2011)
6. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2005, vol. 1, pp. 886–893 (2005)
7. Chen, G., Hou, R.: A new machine double-layer learning method and its. In: International Conference on Mechatronics and Automation ICMA 2007, pp. 796–799 (2007)
8. Wang, Z., Yoon, S., Hong, C., Park, D.S.: A novel SVM based pedestrian detection algorithm via locality sensitive histograms. In: Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV), vol. 2, p. 1, The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp) (2014)
9. Wang, Z., Yoon, S., Xie, S.J., Lu, Y., Park, D.S.: A high accuracy pedestrian detection system combining a cascade AdaBoost detector and random vector functional-link net. Sci. World J. **2014**, 7 p. (2014). doi:10.1155/2014/105089. Article ID 105089
10. Cuyang, W., Wang, X.: Joint deep learning for pedestrian detection. In: 2013 IEEE International Conference on Computer Vision (ICCV), pp. 2056–2063. IEEE (2013)
11. Zeng, X., Ouyang, W., Wang, M., Wang, X.: Deep learning of scene-specific classifier for pedestrian detection. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014 Part III. LNCS, vol. 8691, pp. 472–487. Springer, Cham (2014). doi:10.1007/978-3-319-10578-9_31
12. Fukui, H., Yamashita, T., Yamauchi, Y., Fujiyoshi, H., Murase, H.: Pedestrian detection based on deep convolutional neural network with ensemble inference network. In: 2015 IEEE Intelligent Vehicles Symposium (IV), pp. 223–228. IEEE (2018)
13. https://www.multitel.be/projets/boss/. Accessed 05 Sept 2017

14. Cong, D.N.T., Achard, C., Khoudour, L.: People re-identification by classification of silhouettes based on sparse representation. In: 2010 2nd International Conference on Image Processing Theory Tools and Applications (IPTA), pp. 60–65. IEEE (2010)
15. Cong, D.N.T., Khoudour, L., Achard, C., Flancquart, A.: Adaptive model for object detection in noisy and fast-varying environment. In: Image Analysis and Processing–ICIAP 2011, pp. 68–77 (2011)
16. Coniglio, C., Meurie, C., Lézoray, O., Berbineau, M.: People silhouette extraction from people detection bounding boxes in images. Pattern Recog. Lett. **93**, 182–191 (2017)
17. University of Maryland.: ViPER: The Video Performance Evaluation Resource (2003). http://viper-toolkit.sourceforge.net/docs/. Accessed 24 June 2017
18. Quinteros, D., Velastin, S.A., Acuna, G.: Characterisation of the spatial sensitivity of classifiers in pedestrian detection. In: 6th LatinAmerican Conference on Networked Electronic Media, Medellin, Colombia (2015)