ISBN 978-84-15914-12-9

An evaluation of Cascade Object Detector and Support Vector Machine methods for People Detection using a RGB-Depth camera located in a zenithal position

Eduardo López-de-Celis[1], Óscar García-Olalla[2], Maite García-Ordás[3], Enrique Alegre-Gutiérrez[4]
Escuela de Ingenierías Industrial e Informática, Universidad de León, Campus de Vegazana s/n, 24071 León, Spain, eloped00@estudiantes.unileon.es[1], ogaro@unileon.es[2], mgaro@unileon.es[3], ealeg@unileon.es[4]

Abstract

This project solves the problem of people detection using an RGB-Depth camera from a zenithal position. The detection process has been implemented for binary (head – no head) and multiclass approaches (short hair head, long hair head, ponytail and shoulders-no head). For this task, Histogram of Oriented Gradients (HOG) demonstrates to be a better feature descriptor than Local Binary Patterns (LBP). In the classification step, two models have been evaluated: SVM and Cascade Object Detector. Our experiments shown the better performance of SVM.

Keywords: head detection, Histogram of Oriented Gradients, Local Binary Patterns, Cascade Object Detector, Support Vector Machine

1 Introduction

The main goal of this paper is to solve the problem of people detection passing under an RGB-Depth camera, which is located in a zenithal position. Several works in the recent years have been dealing with this problem too.

In 2013, Tian et al [13], proposed a method, which aims to solve this problem extracting HOG features on depth maps obtained with an RGB-D camera and classifying them with Support Vector Machine (SVM). To delete unnecessary information, they process the images with a movement detector, removing the background. In the work of Castrillón-Santana [5], a similar evaluation was carried out but including tracking and counting of the people passing under the RGB-D camera.

Background subtraction is a very common preprocess for people detection algorithms, which just need extracting features of head and shoulders. Some experiments have shown the advantages of this step [7, 14].

In 2014, Liu et al proposed a method based on Hough circles to detect head contours, achieving very interesting results in terms of time and accuracy [6].

A different approach using frontal view images was proposed by Park and Moon [9]. In their work, they employ new features that encode the similarity or difference between color histograms and oriented gradient histograms to detect the position of the people.

Most of the researchers of these previous papers agree that the head and shoulders shape is the most invariant part of the human body. We have proposed two methods to describe these parts: HOG (shape) and LBP (texture). As the people will be in movement under the RGB-D camera, a movement detector was made, removing all the non-necessary data, and a background subtraction was done, acting as a height filter with a concrete threshold.

The proposed software has been deployed in a Raspberry Pi B 2 with Raspbian OS. We have developed our methods using Matlab, Python, and images processing libraries, like OpenCV and Scikit for the implementation of HOG, LBP, Cascade Object Detector and SVM. The RGB-D camera was a Kinect 360.

The rest of the paper is organized as follows. In section 2 we show the people detector algorithms with more detail. Section 3 introduces the dataset evaluated. Our experiments are shown in section 4 and finally, conclusions are discussed in section 5.

2 People Detector

This section is related to the building of the head and shoulders classifier. Different approaches have been evaluated. Figure 1 shows the general process.

We are going to detail the whole process, explaining which parts of the approaches are common, and which not.

ISBN 978-84-15914-12-9

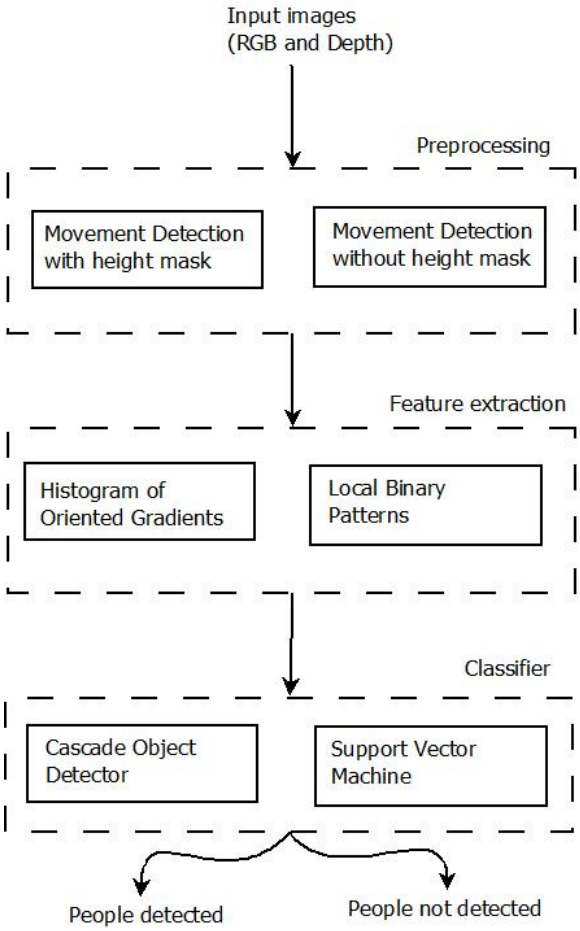

Figure 1: General flow diagram of our work

2.1 Preprocessing images

Our main goal is detect people by their heads and shoulders; in this context, it seems reasonable to think that all the people will be in movement, as they are constantly passing inside and outside a classroom.

For this reason, a movement detector is applied to all the images in two different processing threads: we call them processing thread with mask and processing thread without mask, making a reference to a height filtered mask. The movement detector is the one developed by Stauffer and Grimson [10].

2.1.1 Processing thread without mask

This is the simplest one. Given the RGB image, we just convert it to grayscale image and then, we obtain its movement regions with the movement detector.

The movement detector employed is a foreground detector system. This system compares the grayscale image with a background model to distinguish between foreground and background pixels. After this, a foreground mask is applied.

2.1.2 Processing thread with mask

With the depth image, where its maximum pixel value corresponds to the floor and its minimum (zero) to the ceiling, we compute a filtered height mask: taking the maximum pixel value, we divide it into three parts; the first part will correspond to the upper objects, where the heads and shoulders are located.

Then, we apply this filtered height mask on the grayscale image obtained through the RGB image, removing all the objects that are not in the upper part (approx. from 1.60 meters, in order to dismiss unnecessary information. See Figure 2).

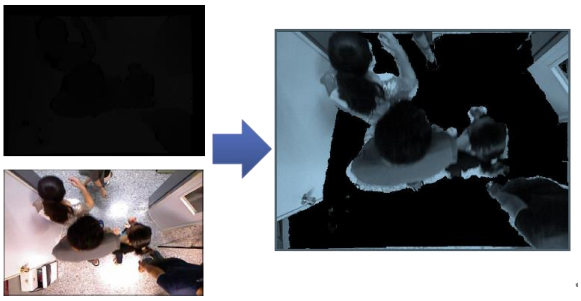

Figure 2: Height Filtered Mask applied to an image. Using the RGB and Depth images as input, we compute a height filter with the first one and we apply it to the RGB image, passed to grayscale, for removing all the objects that are not in the upper part of the image.

Finally, we use the movement detector to the filtered grayscale image in order to get the movement regions. The final result can be seen in Figure 3.

Figure 3: Movement detector applied to an image. White regions correspond to objects in movement.

ISBN 978-84-15914-12-9

This approach is better than the one without height mask in the sense that we remove some unnecessary data from the original image.

2.2 Image Description

Once the images are preprocessed, both approaches come to a feature extraction. As we want to describe heads and shoulders from a zenith position, we have considered, separately, two different descriptors: Local Binary Patterns (LBP), for extracting texture features; and Histogram of Oriented Gradients (HOG) for getting shape information.

2.2.1 Histogram of Oriented Gradients

HOG extracts shape information from the image calculating its point gradients yielding a vector whose directions indicate the maximum variation of intensities.

Given an image, a sliding window of 16x16 size, divided in four cells of size 8x8 pixels, goes over it, with an overlapping of one cell. For each cell, 64 gradient magnitudes are extracted and inserted into a 9-bin histogram. The range of the histogram depends on the use of signed gradients, [0, 360), or unsigned gradients, [0, 180). Each gradient contributes to form the two closest bins.

Then, contrast normalization is carried out. There are two ways to do this task: normalizing the histogram of each cell with L-1 norm (see equation 1) and concatenating the histograms of all of them, or concatenating the histograms of the four cells of the block and normalizing the result with the L-2 norm (see equation 2). Both solutions are frequently used for this task.

$$L1-norm: v \rightarrow v/(\| v \|_1 + \varepsilon) \qquad (1)$$

$$L2-norm: v \rightarrow v/\sqrt{\| v \|_2^2 + \varepsilon^2} \qquad (2)$$

The epsilon is a constant added in order to avoid divisions by zero (equation 1) or negative square roots (equation 2).

Finally, the descriptor is built by the concatenation of the histograms of the blocks. Shape information of a head is shown in Figure 4.

In our case, we have proven the Dalal-Triggs approach [3] and the UOCCTI one [11], available in the VLFeat and Scikit image processing libraries.

2.2.2 Local Binary Patterns

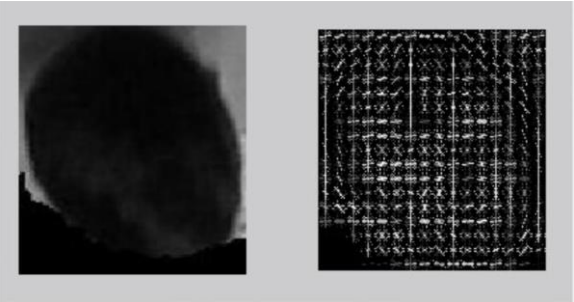

Figure 4: HOG features (right) of a head (left). The vectors describe the shape of the head.

LBP is a very well-known method to extract texture information on images. In a neighborhood inside the image (we have established eight neighbors), the gray scale level of the central pixel is compared to each of the gray scale levels of its eight neighbors. If the center value is higher or equal than the neighbor's value, a one is inserted, otherwise, a zero. Finally, all the values are concatenated, giving a binary string. This binary number is transformed to a decimal value and inserted inside a histogram. The histogram will have 256 different values (2 different values to the power of 8 neighbors). Texture information of a head can be seen in figure 5.

Figure 5: LBP features (right) of a head (left), describing its texture.

2.3 Building the model

Finally, the last step is to build the classifier. We have done this in two different ways: on one hand, using a Cascade Object Detector and, on the other hand, a Support Vector Machine (SVM).

2.3.1 Cascade Object Detector

The Cascade Object Detector uses the Viola-Jones algorithm [4] to detect objects. Many platforms provide trained Cascade Object Detectors of nose, frontal face, upper-body, etc. However, our main goal is to detect heads and shoulders from an upper view, so we had to train our own detector. For doing this task, we provide two sets of negative and positive samples and we train the Cascade Object Detector.

ISBN 978-84-15914-12-9

The training process [12] consists on stages of an ensemble of weak learners, trained using boosting. On every stage, a sliding window goes over every input image and classifies this region as positive or negative. If the region is labeled as positive, this region goes to the next stage, rejecting the negative ones. The process finishes when a certain number of stages, established by the users, are reached, or when there are no more negative samples for the next stage. The trained detector will be returned as an XML file.

One advantage of this mechanism is that it extracts automatically the HOG or LBP (not riu2 [5]) features from the images when it trains and tests the detector, making the process faster.

2.3.2 Support Vector Machine

SVM is one of the most common used methods for supervised learning. Given a set of training samples, we can label the samples and train a SVM to build a model that can predict the class of a new sample. In order to obtain a good classification, SVM can support different kernel functions: linear, Gaussian, polynomial, hyperbolic tangent, etc., allowing the data representation in different spaces (1-D, 2-D…N-D) and helping the mechanism to establish a good decision frontier.

3 Dataset

The employed dataset is the one given by Castrillón-Santana et al [2], which is very similar to our needs: it contains people, passing through a door, and recorded from a zenithal position using an RGB-D camera (see Figure 6).

As we made a supervised training, we had to label all the samples. For this task, we used the Training Image Labeler tool, provided by Matlab. It is part of the Image Processing module, inside the Artificial Vision System toolbox. Labelling with this tool, we obtain an m-file with the following information: path where the image is and an array with the coordinates (x,y) of the upper left corner of the region of interest (ROI) and its dimensions (width and height).

The dataset has been labelled in two different ways: for single-class classification and for multiclass classification. The single-class classification (see Figure 7) splits the dataset in two subsets: positive (heads) and negative (no heads). The multiclass classification is more challenging and separates the dataset in four subsets (see Figure 8): short hair heads, long hair heads, ponytail and shoulders (for Cascade Object Detector) or no head (for SVM).

Except for the long hair cluster, all the figure 7 and 8 clusters belong to figure 6.

Figure 6: Sample of the dataset

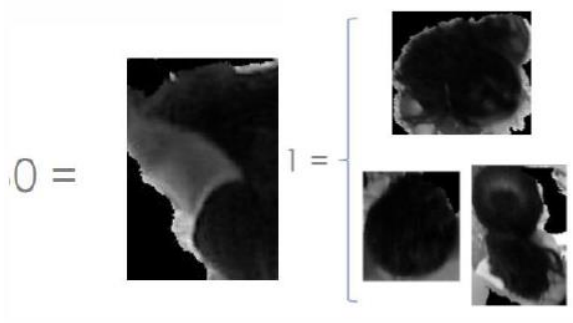

Figure 7: Binary division of the dataset. The zero class represents regions around heads, inside the movement region. The one class contains heads.

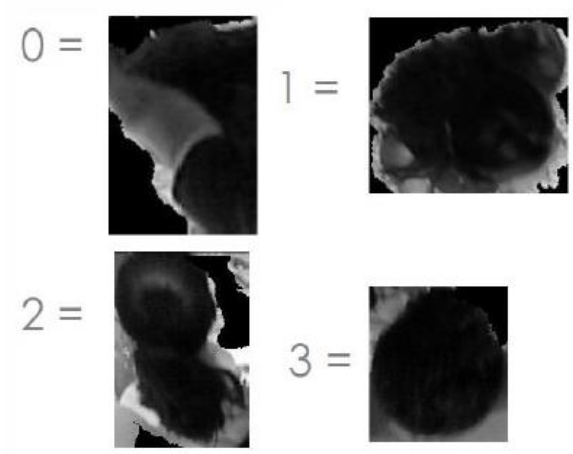

Figure 8: Multiclass division of the dataset. The zero class represents regions around heads (for SVM) or shoulders (for Cascade Object Detector). Class one is for heads with long hair, class two is for heads with ponytail and class three is for heads with short hair.

ISBN 978-84-15914-12-9

4 Evaluation

The evaluation has been made as follows: first, the Cascade Object Detector was trained with HOG and LBP features extracted from the multiclass dataset; later, the SVM, with the same features, but with binary and multiclass classification approaches.

In our testing process, the multiclass method has always been *one vs. the rest*: each class is taken as positive, leaving the rest as negative.

The tables that show the test results use the following statistics:

- True Positive Rate (TPR): correct classification rate. A positive sample has been classified correctly as positive.
- True Negative Rate (TNR): correct rejection rate. A negative sample has been classified correctly as negative.

4.1 Using Cascade Object Detector

This model has always been trained with a dataset preprocessed with the processing thread without mask. By employing one approach vs. the rest multiclass method, four different models, one for each class, were trained. The number of images used for each case (both HOG and LBP) was as follows:

- Short Hair Heads (SHH): 5037 train images, 2569 test images.
- Long Hair Heads (LHH): 5035 train images, 1710 test images.
- Ponytail: 5032 train images, 2817 test images.
- Shoulders: 4647 train images, 2176 test images.

Two important parameters for the classification model are the scale factor and the merge threshold. The scale factor (SF) scales the detection resolution incrementally between a min and max sizes, while the merge threshold (MT) value acts as a threshold merging multiple area detections around the target.

The best results for the Cascade Object Detector with HOG are shown in Table 1 and with LBP in Table 2. As it can be seen, the first descriptor returns better classification results than the second one for this project. A comparative graphic is shown in Figure 9.

Table 1: Best results for the Cascade Object Detector with HOG

Class	SF	MT	TPR	TNR
SHH	1.005	4	93%	88%
LHH	1.005	4	92%	91%
Ponytail	1.005	8	100%	93%
Shoulders	1.005	20	92%	67%

Table 2: Best results for the Cascade Object Detector with LBP

Class	SF	MT	TPR	TNR
SHH	1.005	3	80%	72%
LHH	1.002	3	89%	88%
Ponytail	1.005	7	86%	84%
Shoulders	1.01	6	66%	65%

For obtaining the statistics, we calculate the intersection of the areas of the detected region and the real region, where we know that there is an object of interest. If the intersection is above 50%, we count the detection as TPR.

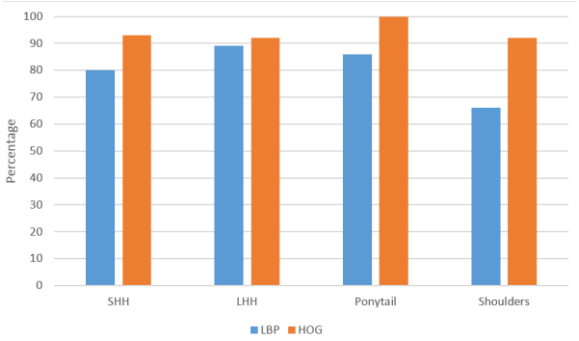

Figure 9: Graphic that compares the TPR percentages of HOG (orange) and LBP (blue) with the Multiclass Cascade Object Detector.

Despite this facts, the results obtained with both cases were not as good as we expected (we expect 97% of correct classification in lab environment). This is the reason why we changed the classification approach to SVM and we simplify the problem adding a new approach: binary classification (head – no head), and leaving the shoulders detection in the multiclass approach.

4.2 Using Support Vector Machine

SVM has been used for classification with the dataset preprocessed with both processing threads already described. Moreover, the binary and multiclass classification approaches have been applied. At this point, the importance of having balanced classes started to take it into account, that is, we started to make tests with the same number of positive and negative images; and unbalanced, with different number of them. Knowing this, the number of images used for each case (both HOG and LBP) was as follows:

- Training with images processed with mask: for balanced classes, 2439 train images, 758 test images; for unbalanced classes: 1144 train images, 758 test images.
- Training with images processed without mask: for balanced classes, 2174 train images, 608 test

ISBN 978-84-15914-12-9

images; for unbalanced classes, 1144 train images, 608 test images.

The best results for a binary classification with HOG and LBP are shown in Tables 3 and 4, respectively.

According to the results, the classification improves noticeably employing SVM with binary classification, reaching the desired percentages. Between HOG and LBP, this last one works worse than HOG. It is from this time when we decide not to continue using LBP. A comparative graphic is shown in Figure 10.

Table 3: Best results for the SVM binary classification with HOG

Kernel	Mask	TPR	TNR
Linear	Yes	97.9%	96.6%
Linear	No	100%	93.9%
Gaussian	Yes	98.5%	97.6%
Gaussian	No	98.2%	94.8%

Table 4: Best results for the SVM binary classification with LBP

Kernel	Mask	TPR	TNR
Linear	Yes	81.5%	90.3%
Linear	No	91.4%	84.7%
Gaussian	Yes	89.8%	90.5%
Gaussian	No	92.6%	86.1%

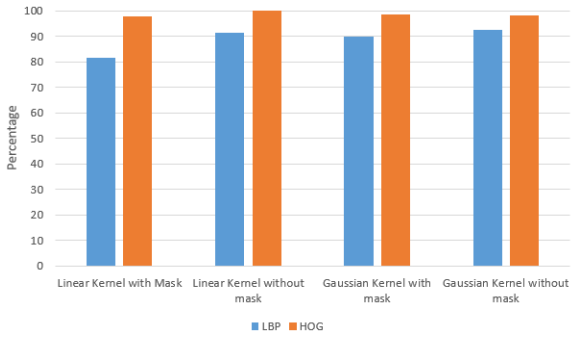

Figure 10: Graphic that compares the TPR percentages of HOG (orange) and LBP (blue) with the Binary SVM classification

In addition, between the approach with mask and without mask, with mask approach with Gaussian kernel works better, because the results returned with this features are high for both True Positive Rate and True Negative Rate.

Because of this good results, we wanted to try the multiclass approach again, but with SVM, and replacing the shoulders class by the no-head class. The obtained output resulted to be equally bad or worse than the results obtained with the multiclass Cascade Object Detector. They can be seen in Table 5.

Table 5: Best results for the SVM multiclass classification with HOG

Kernel	Mask	TPR	Error
Linear	Yes	71.64%	28.36%
Linear	No	68.17%	32.83%
Gaussian	Yes	74.87%	25.13%
Gaussian	No	71.59%	28.41%

5 Conclusions

This work solves the problem of detecting people by their heads using a RGB-D camera from a zenithal position. Using the dataset of Castrillón-Santana [2], we preprocessed the images using two different ways: applying a height mask, and not applying it. Then, two different descriptors have been extracted: HOG and LBP, with two different training model mechanisms: Cascade Object Detector and SVM. Moreover, we have tried to add more complexity to the problem by making a multiclass classification, without success due to the lack of variety in the samples. At the end, better results have been obtained using HOG descriptor and SVM binary classification with images with the height mask applied and Gaussian kernel: 98.5% of samples correctly classified as positive and 97.6% of the samples correctly classified as negative.

Acknowledgements

This work has been developed thanks to the funds received by the Universidad-Empresa challenge first prize, from the Junta de Castilla y León. The prize was won by the Artificial Vision and Pattern Recognition Group (VARP) from the University of Leon, because of its proposal, VisPCounter.

References

[1] Castrillón, M., Lorenzo, J. and Hernández, D, (2014), Conteo de personas con un sensor RGBD comercial, Revista Iberoamericana de Automática e Informática Industrial (RIAI), Vol: 11, Issue: 3.

[2] Castrillón, M., Lorenzo, J. and Hernández, D., (2014), People semantic description and re-identification from point cloud geometry, International Conference on Pattern Recognition, Stockholm, Sweden.

ISBN 978-84-15914-12-9

[3] Dalal, N. and Triggs, B., (2005) Histograms of Oriented Gradients for Human Detection, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA.

[4] Jones, Viola, Paul and Michael (2001), "Rapid Object Detection using a Boosted Cascade of Simple Features", Proceedings of the 2001 IEEE Computer Society Conferene on Computer Vision and Pattern Recognition, volume: 1, pp.511-518.

[5] Liao, S., Zhu, X., Lei, Z., Zhang. L. and Li, S. (2007), Learning Multi-scale Block Local Binary Patterns for Face Recognition, International Conference on Biometrics (ICB), pp. 828-837.

[6] Liu, H., Quian, Y. and Liu, S. (2014), Detecting Persons using Hough Circle Transform in Surveillance Video, Chinese Academy of Science, Beijing, China.

[7] Mukherjee, S. and Das, K. (2013), A Novel Equation based Classifier for Detecting Human in Images, Don Bosco College of Engineering and Technology, Assam Don Bosco University, Guwahati, Assam, India.

[8] Ojala, T., Pietikäinen, M. and Mäenpää, T. (2001), A generalized Local Binary Pattern Operator for Multiresolution Gray Scale and Rotation Invariant Texture Classification, University of Oulu, Finland.

[9] Park, L. and Moon, J. (2013), Exploiting Global Self Similarity for Head-Shoulder Detection, World Academy of Science and Technology, Vol: 7, No: 4.

[10] Stauffer, C., & Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking. In Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on. (Vol. 2). IEEE.

[11] The authors of VLFeat, (2007-13), Histogram of Oriented Gradients (HOG) features, available online: http://www.vlfeat.org/api/hog.html [06/2015]

[12] The Mathworks Inc., (1994-2015), Train a Cascade Object Detector, available online: http://es.mathworks.com/help/vision/ug/train-a-cascade-object-detector.html [06/2015]

[13] Tian, Q., Zhou, B., Zhao, W., Wei, Y. and Fei, W. (2013), Human Detection using HOG Features of Head and Shoulder Based on Depth Map, North China University of Technology, Beijing Urban Engineering Design and Research Institute and Systems Engineering Research Institute, Beijing, China.

[14] Ye, Q., Gu, R. and Ji, Y. (2013), Human detection based on motion object extraction and head-shoulder feature, University of Posts and Telecommunications, Beijing, China.