ISSN 2007-9737

Single-Camera Automatic Landmarking for People Recognition with an Ensemble of Regression Trees

Karla Trejo, Cecilio Angulo

Universitat Politècnica de Catalunya, Barcelona,
Spain

karla.andrea.trejo@upc.edu, cecilio.angulo@upc.edu

Abstract. Active Appearance Model (AAM) is a computer vision procedure for statistical matching of object shape and appearance between images. A main drawback in this technique comes from the construction of the shape mesh. Since landmarks must be manually placed when training shapes, AAM is a very time consuming procedure and it cannot be automatically applied on new objects observed in the images. An approach for automatic landmarking of body shapes on still images for AAM training is introduced in this paper. Several works exist applying automatic landmarking on faces or body joints. Here, we explore the possibility to extend one of these methods to full body contours and demonstrate it is a plausible approach in terms of accuracy and speed measures in experimentation. Our proposal represents a new research line in human body pose tracking with a single-view camera. Hence, implementation in real-time would lead to people being recognized by robots endowed with minimal vision resources, like a webcam, in human-robot interaction tasks.

Keywords. Active appearance model, statistical matching, people recognition, body shape, automatic landmarking, human-robot interaction.

1 Introduction

Recognition is one of the core pursuits in computer vision research [21, 2, 32]. This task attempts to attach semantics to visual data such as images or video. Object recognition is an important and recurring subtopic where models are built to recognize object categories or instances [20, 24, 4]. Yet another subtopic currently getting quite popular is people recognition [31]. People recognition comprises two major interests: attaching identities to pictures or video, and building descriptions from visual data of people behaviour. These descriptors lead to a variety of tasks which can be performed based on those premises, like face recognition [23, 17, 33], pose estimation [26], activity recognition [7, 28], and people tracking [1], among others.

Properly identifying humans includes several difficulties to be overtaken. Images or video of people can show a high degree of variability in shape and texture. Appearance variations are due to differences between individuals, deformations in facial expression, pose and illumination changes. One should also take into account visual perception parameters such as resolution, contrast, brightness, sharpness, and color balance [5, 35, 27, 6]. Model-based techniques represent a very promising approach where a model representing an identity of interest is matched with unknown data. This kind of techniques have shown to be able of entirely describing facial characteristics in a reduced model, extracting relevant face information without background interference [16, 15, 8].

Active Appearance Models (AAMs) [9] are generative models of a certain visual phenomenon. Although linear in both shape and appearance, overall, AAMs are nonlinear parametric models in terms of pixel intensities. Fitting an AAM to an image consists of minimizing the error between the input image and the closest model instance. Frequent applications of AAMs include medical image interpretation, face recognition and tracking. Nevertheless, a major issue lies in the construction of the 2D shape mesh as landmarks must be placed by hand on the training images, which is

doi: 10.13053/CyS-20-1-2365

a very long and time consuming process to carry out.

The task of constructing AAMs without hand-marking the mesh is called *Automatic AAM* and the process to automatically localize the vertexes in that mesh is known as *Automatic Landmarking*. Several approaches have been performed to achieve this task on images of human faces, either static or dynamic, combining feature descriptors and predictors [3, 34, 29]. However, state-of-the-art methods employ upgraded versions of AAMs combined with decision-tree learning algorithms, leading to outstanding face alignment results [11, 18, 22]. RGB-Depth sensors technology also contributes to widen the method's scope [14], but the aim in the present work is to cover only 2D data and analyze the accuracy of less computationally expensive instances.

Besides, there exists a vast body of work on the estimation of articulated human pose from images and video. Recent studies of 2D pose estimation from still images reveal prominent results using regressors [12, 30, 25]. Hence, the approach introduced in [22] for face alignment has been selected to be implemented in the automatic landmarking of body contours on still images. The algorithm employs a cascade of regressors combined with decision-tree learning, conforming a very suitable combination for our new specific target on body contours as it contains the latest advanced methodologies of automatic AAMs and human pose estimation.

The main contribution of this article is to demonstrate that facial automatic landmarking approaches can be extended to body contours as well. Hence, the objective for the presented work is to obtain solid evidences that algorithms from the current literature can be implemented and adequate to perform the automatic landmarking task of still images on body contours with acceptable accuracy. Moreover, system execution will be validated for real-time by means of a speed test. Landmarking of still images is obtained without any motion capture data [19] or multiple camera views [13], just by annotating a really small dataset. In this form, AAMs methods application will be enlarged in the people recognition and tracking domain.

Experimentation consists in selecting four representative images - from three out of four different subjects, 12 images in total - from our dataset of 46 images to be manually landmarked, in order to feed a body shape predictor trainer. We are not considering our fourth subject in the shape predictor training. The output of this trainer gives us a general shape model for human bodies that, combined with a people detector, will be used to fit in and automatically landmark the body contours of any new image presented, either from new frames of the three trained subjects or the completely new fourth subject. Results from this experiment will confirm that, with proper adjustments, face alignment algorithms can be exported to human bodies with a very low computational time for testing, which fosters real-time implementation for robots equipped with a single camera.

The rest of this paper is organized as follows. At the first stage of the procedure, the people detector and its training are detailed in Section 2. In Section 3 a complete description is depicted about the training process for the body shape predictor as the second stage for our system. The third and last stage is presented in Section 4, where the automatic landmarking system is ready to go, experiments are conducted and results analyzed. Finally, conclusions and future research lines can be found in Section 5.

2 People Detector

The first phase of the overall proposed system is people detector training. It will be implemented using the `imglab` tool from the *dlib C++ Library*[1]. From a set of images of different subjects obtained using webcam streaming, a subset of images is used to train the people detector, and they will not be used in further stages of the system.

Hence, from a set of 46 images provided by 4 different subjects from a webcam streaming with a 640×480 standard size, 26 of them were taken only to train the people detector. As a matter of fact, none of the 46 images duplicate in other subsets. Each dataset has its own share so as to demonstrate the algorithm is not influenced by

[1] http://dlib.net/

preferences for a particular subject. Using these 26 images -from Subject 2, Subject 3, and Subject 4- as an input in the *imglab* tool we were able to annotate our dataset with red bounding boxes over the whole body of each subject and label them as *body_detected* (see Figure 1). There were no other subjects or any object with an anthropomorphic form within the images. Therefore, no crossed bounding boxes were necessary to indicate false positives which *dlib C++* code should ignore when performing the detections.

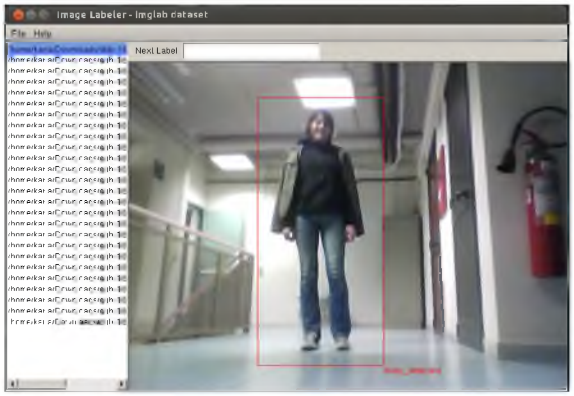

Fig. 1. Annotating images with the *imglab* tool to train the people detector system

2.1 Training

Several tests were conducted to obtain suitable training parameters for the best possible detection results. Moreover, variables like the number of training images and tightness of the bounding box were taken into account.

The most important training parameters for the people detector system are shown in Table 1 with the selected values. *dlib C++ Library* goes through the steps to train a kind of sliding window object detector as the one published in [10] and summarized in Figure 2, applying HOG as feature descriptor and linear SVM as baseline classifier. Consequently, the trainer has the usual SVM's *C* hyper-parameter (see Table 1). The most favorable value for *C* was empirically found by analyzing the performance of the trained detector on a test set of new images, all of these images are not used in the training stage. In general, higher values for *C* encourages it to fit the training data better but might lead to overfitting.

Table 1. Default parameter values set by the example program versus chosen values after running some tests

Parameter	Deafault Value	Selected Value
C	1	0.65
epsilon	0.01	0.01
target-size	80×80	100×100
upsample	no	no
flip	no	yes

The trainer keeps running until the 'risk gap' is small enough (less than the *epsilon* value in Table 1). For most problems a value in the range from 0.1 to 0.01 is plenty accurate. Although smaller values prompt the trainer to solve the SVM optimization with a higher accuracy, it will take longer to train. The term *target-size* in Table 1 refers to the size in pixels of the sliding window detector.

For this training data in particular, the *upsample* function does not improve the results. Acting as a pre-processing step, *upsample* increases the size of the images by a factor of 2 allowing us to detect smaller features on the target object. This is not our case because we are focused just on human silhouettes, which already are quite large to detect in any circumstance.

Since human bodies are usually left-right symmetric we can increase our training dataset by adding mirrored versions of each image with the *flip* option. This step is really useful as it doubles the size of our training dataset from 26 to 52 images, improving the results significantly. Another remarkable detail was to train the bounding boxes a bit loose from the bodies. if trained too tight, detections tend to be partial. Under final corrections, our people detector started with detections like the one shown in Figure 3a, but with proper adjustments achieved results as in Figure 3b. Bodies with complicated poses are difficult to entirely include in the detection box (Figure 3c). Nevertheless, setting the parameters to their final values in Table 1 led to obtaining better detections of this particular instance (Figure 3d).

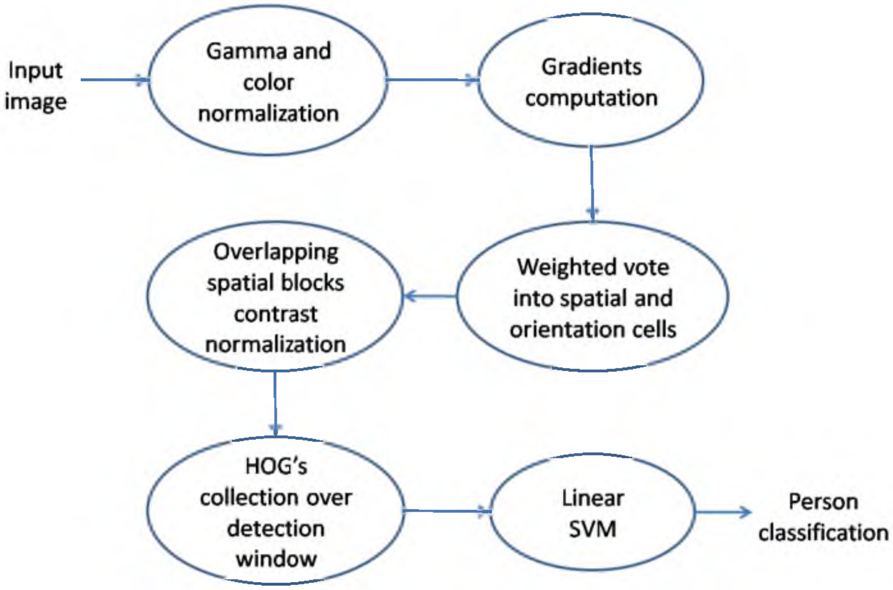
Input image
Gamma and color normalization
Gradients computation
Weighted vote into spatial and orientation cells
Overlapping spatial blocks contrast normalization
HOG's collection over detection window
Linear SVM
Person classification

Fig. 2. An overview on the feature extraction and object detection algorithm by Dalai and Triggs. Histogram of Oriented Gradients feature vectors is extracted from the detector window tiled with a grid of overlapping blocks. This detection window is scanned across the image at all positions and scales. Finally, the combined vectors are fed to a linear Support Vector Machine for object/non-object classification

3 Body Shape Predictor

Functions in the *dlib C++ Library* implement methods described in [22], where they show how an ensemble of regression trees can be used to estimate face landmark positions directly from a sparse subset of pixel intensities, achieving real-time performance with high quality predictions. Similarly, we have trained a shape model by estimating body landmark locations to evaluate and demonstrate whether the algorithm can be extrapolated to body contours as well.

The directory for the body shape predictor contains a training dataset and a separate testing dataset. The training dataset consists of 4 images from Subject 1, Subject 2, and Subject 3; while the testing dataset comprises the same images as before, but mirrored. Every image is annotated with rectangles that bound a human body along with 180 landmarks on each body shape. The objective is to use this training data for learning to identify the position of landmarks on human bodies in new images. Once the shape predictor has been trained, it is tested on previously unseen data, the testing dataset.

3.1 Training

In this study, we assume to work on a very small training dataset because (i) there is no available landmarked datasets of 2D human body contours, so we built the training dataset ourselves by hand, and (ii) it is unrealistic to assume large datasets of hand-made landmarked people. Training data is loaded from an XML file having a list of the images in each dataset and also containing the positions of the body detection boxes and their landmarks (called *parts* in the XML file) as shown in Figure 4. A total of 180 landmarks were distributed by hand, quite close to each other (10 pixels away, approximately), so that the body shape predictor trainer could be more efficient generating the shape model.

The body shape predictor trainer has several parameters to be tuned. A general overview of this training is depicted in Figure 5, for a more detailed explanation we refer to [22]. Default values

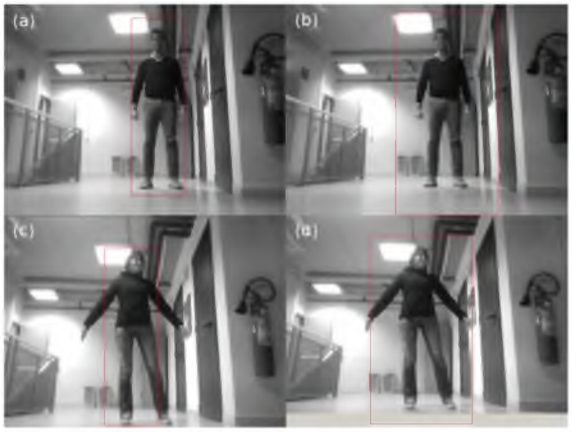

Fig. 3. People Detector results. (a) Natural standing pose, the detection box is very tight and the subject's left hand is missing. (b) Same image as (a) but with final selection of parameters and variables, although the detection box is now too loose for this target to complete the detection. (c) Subject stretches spreading her limbs, the detection box is not able to contain both arms nor her left foot. (d) Same image as (c) but with final selection of parameters and variables, the detection box only misses the tip of the subject's right hand

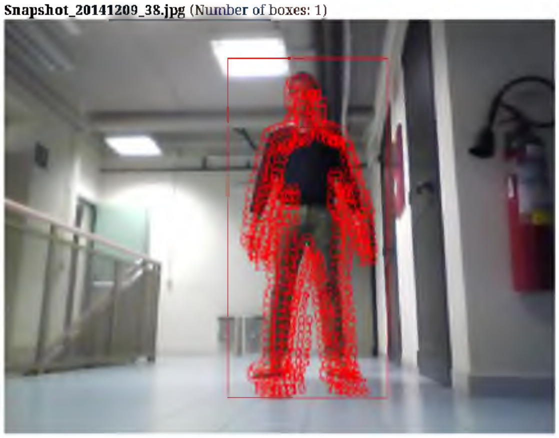

Fig. 4. Image from the training dataset with the target body contained in a detection box and its shape landmarked by 180 enumerated points

from the *dlib C++ Library*'s function are used in our experimentation, except for higher *oversampling* value to effectively boost the training set size, which is really small in our experiments; smaller *nu* value to reduce the capacity of the model by explicitly increasing the regularization; and smaller value for the *tree-depth*, to reduce the model as well.

With this information, the algorithm generates the body shape model. The generated model is validated using a measure of the average distance (in pixels) between body landmarks obtained from the shape predictor and where it should be according to the training set, which is called *mean training error*. Yet, the real test lies in how well the predictor performs on unseen data. We test the recently created shape model on a testing set of landmarked mirrored images to avoid hand-marking all over again, obtaining a *mean testing error* which validates in the exact same way as its 'training error' counterpart.

Only one parametrized testing for the body shape predictor has been conducted so far. Setting values to 500 for *oversampling*, *nu* equal to 0.05, and a *tree-depth* value of 2, the shape predictor takes around three minutes for training obtaining a *mean training error* of 9.03275 pixels and a *mean testing error* of 68.8172 pixels. As we proceed to the next stage of the system with this shape model and its characteristics, the overall method accuracy lies in the *mean testing error* value and it will be reflected on the results shown in Section 4.

4 Automatic Landmark Detection

Once the people detector and the body shape model have been trained, it is time to combine them to find frontal human bodies in an image and estimate their pose. The pose takes the form of 180 connected landmarks describing the shape or silhouette of the subject on a given image. Again, this pose estimator was created by using the *dlib C++ Library*'s implementation of [22]. A new testing dataset was built with the 12 remaining images from the initial 46 that were provided for the experimentation, this time with images from all 4 subjects. Although Subject 4 appears in the training stage of the people detector, it was not included in the training of the shape model. Hence, Subject 4 is a *completely unseen* target for our automatic landmarking process, as the people detector stage just helps the overall system to

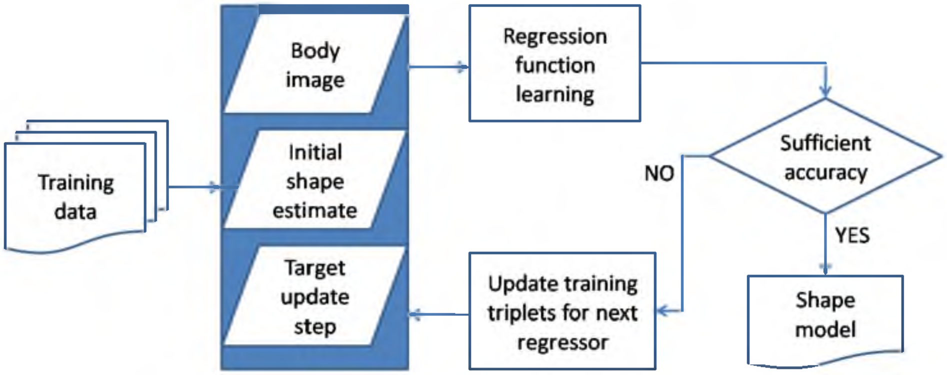

Fig. 5. Flow chart associated to the cascade of regressors training from the algorithm by Kazemi and Sullivan to estimate the position of facial landmarks. Regression functions are learned using gradient tree boosting with a sum of square error loss. The core of each regression function is the tree-based regressors fit to the residual targets during the gradient boosting algorithm. At each split node in the regression tree, a decision is made based on thresholding the difference between the intensities of two pixels

narrow the body search to one area of the image but does not influence the shape fitting procedure.

Now, let's focus our attention on the automatic landmark detection results shown in Figure 6. Throughout the first three frames, the subject maintains a natural standing pose with little variations. It is noticeable from the first (Figure 6a) to the second frame (Figure 6b) how the algorithm is able to refine the detection and make a better fit of the actual position of the head and feet. It keeps improving until the third frame (Figure 6c), where the algorithm actually manages to obtain a good estimation of the real location of the arms and hands of the subject as well. In Figure 6d the algorithm still tries to catch a pose although the subject is partially sided while stepping out from the visible range of the webcam, which is quite remarkable.

These experiments required regular standing poses due to the general objectives of the project which means we do not want to depend on a particular flashy pose that triggers a response from the robot, but rather make the robot notice and identify you just by normally passing by, as in any human notion of another human presence.

However, it was important for us to have some images in the dataset with people also displaying slightly out of the ordinary poses (Figure 7b), so we could reveal the scope and limitations of the algorithm facing this type of cases as well. It is evident we have a rescaling distance issue as seen in Figure 7, but at least the algorithm makes an effort to fit into the spread arms of our subject in Figure 7b. Probably with more similar frames to this one, the algorithm would end refining that pose estimation.

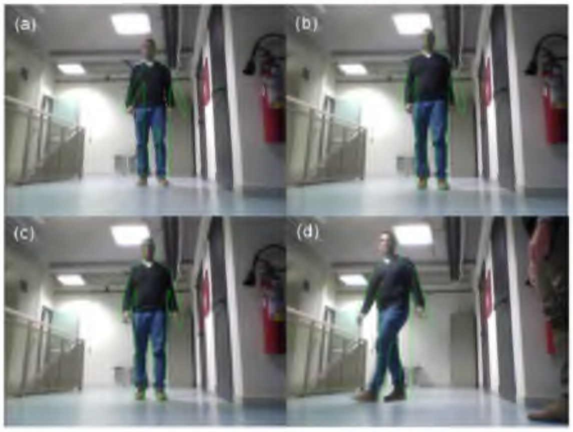

Fig. 6. Automatic Landmarking. Natural standing case with a subject at a regular distance from the webcam

All these automatic landmarking experiments were performed in about 126 milliseconds in a laptop with Intel Core i5-480M and 4GB RAM memory. Depending on the image it could take more or less than this mean test time. However,

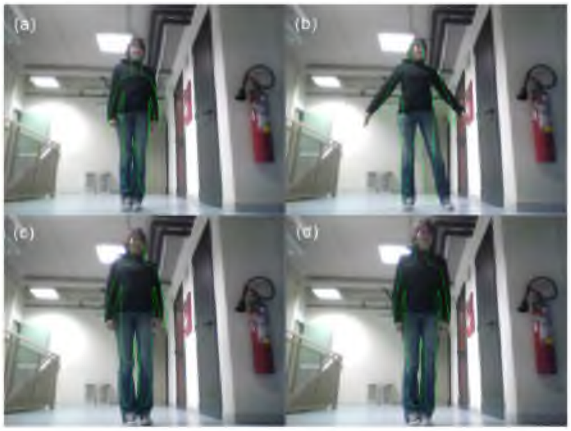

Fig. 7. Automatic Landmarking. Special pose case with a subject approximating the webcam

the procedure in general does not overpass 130 milliseconds, where 80 to 90 milliseconds correspond merely to the people detector stage.

Finally, we present a set of images in Figure 8 from the same subject (the unseen Subject 4). The images leaded to problems in shape prediction due to the training stage of the people detector (Figure 8a). With the final tuning of the parameters we achieved a better detection. Although the box does not reach the top of the head (Figure 8b), it is good enough for the next stage, as the shape predictor starts its landmarking at the top of the forehead instead. While in the process of strategically placing the 180 landmarks on the training data for the shape predictor, we considered hair as a very unstable feature that we can do without and have no major consequences about it. Thus, landmarking starts on the forehead to encircle just the face of the person, becoming a constant human body feature implicitly.

Figure 8c and Figure 8d present the most disappointing outcomes for automatic landmarking. There is no real improvement with respect to the first frame but the process grows worse estimating spread arms instead of spread legs and shortening them; however, it achieves to fit well the left leg. This last fact can be regarded to a significantly slower learning of the unseen body shape. The subject maintains almost the same pose throughout the frames, but apparently this does not help the algorithm to work easier on the image. The problem apparently lies in the subject's clothing rather than the *unseen target* fact as it is almost completely black and there are no considerable disparities in color between body parts, which seems to be difficult to overcome for the algorithm.

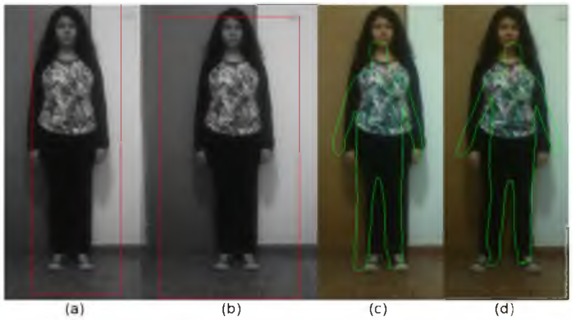

Fig. 8. A difficult case. (a) First test output from the People Detector. (b) Output with selected values for People Detector in its last testing session. (c) Worst automatic landmark detection of the dataset. (d) The automatic landmark detection does not get any better after a second frame of the subject as seen in other cases

5 Conclusions and Future Work

In this paper we have shown an automatic landmarking approach for human body shapes to be carried out on still images from a single camera. It demonstrated a good performance in terms of accuracy and computational time.

The presented system could be really practical and suitable for domestic service robots equipped with basic technological resources due to economic reasons or efficiency purposes. Although there is still much work left to do, these obtained results are quite promising and somehow accomplish the established overall objectives. With the right adjustments it would eventually achieve the necessary performance to be implemented on the primary stages of 2D AAM construction, and subsequently, on people recognition tasks.

Accuracy could be improved with further experimentation and tuning of the shape predictor parameters, besides feeding more landmarked images to the training dataset. A face detector stage could be also useful to reduce our fitting error significantly.

Real-time processing with live video streaming from the webcam certainly is the next imminent step we are looking forward to, as the algorithm clearly possesses this capability and will be likely to have a better chance of success, enabling the system for people tracking tasks.

Future work also involves an interesting contribution we would like to develop on the algorithm: *inter-shoulder distance*. Analogue to what inter-ocular distance achieves on face alignment problems, inter-shoulder distance could become the appropriate solution to rescale distances. *Inter-hip* distance may as well reinforce our body landmarking system and improve upcoming results.

Acknowledgements

Special thanks to Davis King, author of the open source *dlib C++ Library*, who is always willing to assist the community of users.

This research is partially supported by the PATRICIA (TIN2012-38416-C03-01) Research Project, funded by the Spanish Ministry of Economy and Competitiveness. K.A. Trejo acknowledges the received financial support from CONACYT grant.

References

1. **Andriluka, M., Roth, S., & Schiele, B.** (**2008**). People-tracking-by-detection and people-detection-by-tracking. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8.
2. **Ayache, N. & Faugeras, O.** (**1986**). HYPER: A new approach for the recognition and positioning of two-dimensional objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8, No. 1, pp. 44–54.
3. **Baker, S., Matthews, I., & Schneider, J.** (**2004**). Automatic construction of Active Appearance Models as an image coding problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 26, No. 10, pp. 1380–1384.
4. **Belongie, S., Malik, J., & Puzicha, J.** (**2002**). Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 4, pp. 509–522.
5. **Binford, T.** (**1971**). Visual perception by computer. *IEEE Conference on Systems and Control*.
6. **Black, M., Fleet, D., & Yacoob, Y.** (**1998**). A framework for modeling appearance change in image sequences. *IEEE International Conference on Computer Vision*, pp. 660–667.
7. **Blank, M., Gorelick, L., Shechtman, E., Irani, M., & Basri, R.** (**2005**). Actions as space-time shapes. *IEEE International Conference on Computer Vision*, volume 2, pp. 1395–1402.
8. **Blanz, V. & Vetter, T.** (**2003**). Face recognition based on fitting a 3D Morphable Model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 25, No. 9, pp. 1063–1074.
9. **Cootes, T., Edwards, G., & Taylor, C.** (**2001**). Active Appearance Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 23, No. 6, pp. 681–685.
10. **Dalal, N. & Triggs, B.** (**2005**). Histograms of Oriented Gradients for human detection. *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pp. 886–893.
11. **Dantone, M., Gall, J., Fanelli, G., & Van Gool, L.** (**2012**). Real-time facial feature detection using conditional regression forests. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2578–2585.
12. **Dantone, M., Gall, J., Leistner, C., & Van Gool, L.** (**2013**). Human pose estimation using body parts dependent joint regressors. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3041–3048.
13. **Dimitrijevic, M., Lepetit, V., & Fua, P.** (**2006**). Human body pose detection using Bayesian spatio-temporal templates. *Computer Vision and Image Understanding*, Vol. 104, pp. 127–139.
14. **Dopfer, A., Wang, H.-H., & Wang, C.-C.** (**2014**). 3D Active Appearance Model alignment using intensity and range data. *Robotics and Autonomous Systems*, Vol. 62, No. 2, pp. 168–176.
15. **Edwards, G., Cootes, T., & Taylor, C.** (**1998**). Face recognition using Active Appearance Models. In *European Conference on Computer Vision*, volume 1407 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, pp. 581–595.

16. **Edwards, G., Lanitis, A., Taylor, C., & Cootes, T.** (**1996**). Statistical models of face images - Improving specificity. *British Machine Vision Conference*, pp. 765–774.

17. **Edwards, G., Taylor, C., & Cootes, T.** (**1998**). Learning to identify and track faces in image sequences. *IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 260–265.

18. **Fanelli, G., Dantone, M., & Van Gool, L.** (**2013**). Real time 3D face alignment with Random Forests-based Active Appearance Models. *IEEE International Conference and Workshops on Automatic Face and Gesture Recognition*, volume 0, pp. 1–8.

19. **Fossati, A., Dimitrijevic, M., Lepetit, V., & Fua, P.** (**2007**). Bridging the gap between detection and tracking for 3D monocular video-based motion capture. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8.

20. **Grimson, E.** (**1990**). *Object Recognition by Computer: The Role of Geometric Constraints*. MIT Press, Cambridge, MA, USA.

21. **Hoffman, D. & Richards, W.** (**1983**). Parts of recognition. *Cognition*, Vol. 18, pp. 65–96.

22. **Kazemi, V. & Sullivan, J.** (**2014**). One millisecond face alignment with an ensemble of regression trees. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1867–1874.

23. **Kobayashi, H. & Hara, F.** (**1993**). A basic study of dynamic recognition of human facial expressions. *IEEE International Workshop on Robot and Human Communication*, volume 11, pp. 271–275.

24. **Lowe, D.** (**1999**). Object recognition from local scale-invariant features. *IEEE International Conference on Computer Vision*, volume 2, pp. 1150–1157.

25. **Marino, K.** (**2014**). Real time human pose estimation for boosted random forests and pose machines. *Robotics Institute Summer Scholars (RISS) Working Papers*, volume 2, Carnegie Mellon University, pp. 45–49.

26. **Mori, G. & Malik, J.** (**2002**). Estimating human body configurations using shape context matching. *European Conference on Computer Vision. LNCS 2352*, volume 3, pp. 666–680.

27. **Nayar, S., Murase, H., & Nene, S.** (**1996**). Parametric Appearance Representation. In *Early Visual Learning*. pp. 131–160.

28. **Oikonomopoulos, A., Pantic, M., & Patras, I.** (**2009**). Sparse B-spline polynomial descriptors for human activity recognition. *Image and vision computing*, Vol. 27, No. 12, pp. 1814–1825.

29. **Prabhu, U., Seshadri, K., & Savvides, M.** (**2012**). Automatic facial landmark tracking in video sequences using Kalman filter assisted Active Shape Models. In *Trends and Topics in Computer Vision*, volume 6553 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, pp. 86–99.

30. **Ramakrishna, V., Munoz, D., Hebert, M., Bagnell, A., & Sheikh, Y.** (**2014**). Pose machines: Articulated pose estimation via inference machines. *European Conference on Computer Vision*.

31. **Ramanan, D., Forsyth, D., & Zisserman, A.** (**2007**). Tracking people by learning their appearance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 29, No. 1, pp. 65–81.

32. **Turk, M. & Pentland, A.** (**1991**). Eigenfaces for recognition. *Cognitive Neuroscience*, Vol. 3, No. 1, pp. 71–86.

33. **Zhao, W., Chellapa, R., Phillips, P. J., & Rosenfeld, A.** (**2003**). Face recognition: A literature survey. *ACM Computing Surveys*, Vol. 35, pp. 399–458.

34. **Zhou, D., Petrovska-Delacretaz, D., & Dorizzi, B.** (**2009**). Automatic landmark location with a Combined Active Shape Model. *IEEE International Conference on Biometrics: Theory, Applications, and Systems*, pp. 1–7.

35. **Zucker, S.** (**1981**). Computer vision and human perception: An essay on the discovery of constraints. *International Joint Conference on Artificial Intelligence*, volume 2, pp. 1102–1116.

Karla Trejo is a Ph.D. student in the Automatic Control, Robotics and Computer Vision doctoral programme at Universitat Politècnica de Catalunya - BarcelonaTech, Spain. She received her technical degree in 2006 as a Programmer Analyst from Universidad de Colima, Mexico, and in 2011 her Bachelor degree in Mechatronics with a System's Design and Automation specialty from Instituto Tecnológico de Colima, Mexico. She was working at Zona Zero - Tecnologías de Información, Mexico, as an Embedded Systems Developer of the R&D Department when in 2012, the Mexican Government through CONACyT granted her to pursue an M.S. degree at Universitat Politècnica de Catalunya - BarcelonaTech, Spain. As a student of the master program in Automatic Control and Robotics, she was selected to participate in the events held by RoCKIn EU project and became increasingly interested in Computer Vision topics. In 2014, Karla Trejo defended her thesis regarding Active Appearance Models for People Recognition with RGB-D sensors, receiving her M.S. degree. She's currently extending the work and research conducted on this matter to her doctoral studies as a member of the Knowledge Engineering Research Group.

Cecilio Angulo is an Associate Professor of the Department of Automatic Control at the Universitat Politècnica de Catalunya - BarcelonaTech, Spain. He received his M.S. degree in Mathematics from the University of Barcelona, Spain, and his Ph.D. degree in Sciences from Universitat Politècnica de Catalunya - BarcelonaTech, Spain, in 1993 and 2001, respectively. From 2011 he has been serving as Director of the Master degree in Automatic Control and Robotics. He's currently the Director of the Knowledge Engineering Research Group, where he is responsible for research projects in the area of robot learning. Cecilio Angulo is the author of over 250 technical publications. His research interests include cognitive robotics, machine learning algorithms, and computer vision applications.

Article received on 12/09/2015; accepted on 19/11/2015.
Corresponding author is Karla Trejo.