Head Detection with Depth Images in the Wild

Diego Ballotta, Guido Borghi, Roberto Vezzani, Rita Cucchiara
DIEF - University of Modena and Reggio Emilia
Via P. Vivarelli 10, Modena, Italy
{diego.ballotta, guido.borghi, roberto.vezzani, rita.cucchiara}@unimore.it

Abstract

In wild contexts, head detection is a demanding task and a key element for many disciplines of the computer vision community, like video surveillance, Human Computer Interaction and face analysis. In this paper, we introduce a novel method for head detection, that conjugates the classification ability of deep learning approaches and depth maps, a type of infrared-based images useful to achieve reliability in case of light changes or bad light conditions. Moreover, depth data are also employed to deal with one of the traditional problems in object detection task, i.e. the scale of the target object. Two public datasets are exploited: the first one, Pandora, is used to train the deep classifier of face or non-face images; the second one, collected by the Cornell University, is used to perform a cross-dataset test during daily activities in unconstrained environments. Experimental results show that the proposed method overcomes state-of-art performance of methods based only on depth images.

1. Introduction

Human head detection is a traditional computer vision research field, and in last decades many efforts have been conducted to find competitive and accurate methods and solutions. This task is a fundamental step for many research fields based on faces, like face recognition, attention analysis, pedestrian detection, human tracking, to develop real world applications in contexts such as video surveillance, autonomous driving, behavior analysis and so on.

Head detection is a challenging task in wild contexts, due to variations in head appearance and pose and the presence of even dramatic body occlusions, lighting condition changes and complexity of the background. Moreover, the head could be turned away from the camera or in a far-field. Most of the current research approaches are based on images taken by conventional visible-lights cameras – i.e. RGB or intensity cameras – and only few works tackle the problem of head detection in other types of images, like depth images, also known as depth maps or range images. Recently, the interest about the exploitation of depth images is increasing thanks to the wide spread of low cost, ready-to-use and high quality depth acquisition devices, i.e. Microsoft Kinect or Intel RealSense devices. Furthermore, these recent depth acquisition devices are based on infrared light and not lasers, so they are not dangerous for humans and can be used in human environment without particular limitations.

Besides, the use of depth images is required in specific situations, where is important to have available light-invariant systems: for example, this is the case of driver behavior monitoring and investigation tasks, to be performed in all parts of the day and night and with different weather conditions. Finally, literature methods based on depth images have several advantages over method based on 2D information. In particular, 2D based methods suffer the complexity of the background and when subject’s head has not a consistent color or texture.
In this paper, we present a method that is able to detect and localize a head, given a single depth image. The proposed system is based on a deep architecture, created to have good accuracy and to be able to classify head or non-head images. The network is trained on a public dataset, Pandora, and the whole system is tested on another public dataset, namely Cornell University dataset, performing a cross-dataset evaluation. Results confirm the effectiveness and the feasibility of the proposed method, also for real world applications.

2. Related Work

As described above, most of head detection methods proposed in the literature are based on intensity or RGB images. This is the case of the well-known Viola-Jones object detector [19], where Haar features and a cascade classifier (AdaBoost) are exploited to develop a real time and a robust face detector. A specific set of features has to be collected to handle the variety of head poses. Besides, solutions based on SVM [15] and Neural Networks [16] have been proposed to tackle the problem of face or head detection on intensity images.

Only few works present approaches for head detection in depth images. The recent work of Chen et al. [5] presents a novel head descriptor to classify, through a LDA classifier, pixels as head or non-head. Depth values are exploited to eliminate false positives of head centers and to cluster pixels for final head detection. In the work of Nghiem et al. [14], head detection is conducted on 3D data as first step for a fall detection system. This method detects only moving objects through background subtraction and all possible head positions are searched on contour segments. In some works, only head localization is addressed, that is the ability to localize the head into the image, assuming the presence of at least on head in the input image. In these cases, it is frequently supposed that the subject is frontally placed in respect to the acquisition device. This is the case of [7], where depth image patches are used to directly estimate head location and orientation at the same time with a regression forests algorithm. Also in [13] the head center is predicted through a regressive Convolutional Neural Network (CNN) trained on depth frames, supposing the user’s head is present and at least partially centered in frames acquired. In both cases, authors assume also that head is always visible on the top of the moving human body.

Head detection shares some common aspects with face recognition and pedestrian detection tasks and this is why face detection methods often cover the case of human or pedestrian detection. Moreover, most head detector rely on the assumption to find shoulder joints to locate head into the input image.

Gradient-based features such as HOG [6], EOH [12] are generally exploited for pedestrian detection in RGB images. Techniques to extract scale-invariant interesting points in images are also exploited (SIFT, [13]). Other local features, like edgelets [20] and poselets [4], are used for highly accurate human detection. Recently, deep learning based methods are proposed [23] to perform face detection, pose estimation and landmark localization with intensity images. In [22] a human detector based on depth data and a 2D head contour model and 3D head surface model is presented. In addition, a segmentation scheme to extract the entire body and a tracking algorithm based on detection results are proposed.

A multiple human detection method in depth images is presented in [9] and is based on a fast template matching algorithm; results are verified though a 3D model fitting technique. Then, human body is extracted exploiting morphological operators to perform a segmentation scheme. In [8] a method for detecting humans by relational depth similarity features based on depth information is presented: integral depth images and AdaBoost classifier are exploited to achieve good accuracy and real time performance. Shotton et al. proposes in [18] a method based on randomized decision trees to quickly and accurately predict the 3D positions of body joints from a single depth image (also the head is included) and no temporal information are exploited.
3. Face Detection

An overall depiction of the proposed framework is shown in Figure 2. Firstly, given a depth image as input, square patches (head candidates) are extracted. Then, a CNN is exploited as a classifier to predict if a patch contains a human head or not. Only candidates that satisfy certain conditions are considered as real head and their positions are recovered from the patch extracted. The final output of the system are the \( x, y \) coordinates of the face center.

3.1. Depth camera and data pre-processing

Before to describe our method, it is necessary analyzing the acquisition camera we use to collect depth frames due to its effects on our approach. Both Pandora and Cornell University datasets exploit a second generation Microsoft Kinect One device, a Time-of-Flight (ToF) camera. Thanks to infrared light, it is able to measure the distance to an object inside the scene, by measuring the time interval taken for infrared light to be reflected by the object in the scene. Kinect One is able to acquire data in real time (30 fps), with a range starting from 0.5 to 8 meters, but best depth information are available only up to 5 meters. All distance data are reported in millimeters. The sensor provides depth information as a two dimensional array of pixels (depth maps), like a gray-level image. Since each pixel represents the distance in millimeters from the camera, depth images are represented in 16 bit. For this reason, in our system we use 16 bit input images, and depth values are converted in standard 8 bit values (0 to 255) only for visual inspection or representation.

Due to the nature of ToF sensor, noise is frequently present in acquired depth images, and it is visible as dots with zero value (random black spots). Input depth images are pre-processed to remove these values through a median filter. Kinect One is able to simultaneously acquire RGB and depth maps. During patch extraction, kernel are shifted by their half width and height, respectively. So, given a input image of width and height \((w_i, h_i)\), the total number of extracted patches is computed as follow:

\[
\text{\#patches} = \frac{w_i}{(w_k/2)} \cdot \frac{h_i}{(h_k/2)}
\]

where \(w_k, h_k\) is the size of the kernel.

Patches smaller than \(15 \times 15\) pixels are discarded, due to the maximum range of the acquisition device. With this procedure we obtain two major benefits: we do not need to implement any multiple-scale approach, like in [19], so the processing overhead is reduced; the second benefit is that we are able to extract square candidates that contain heads (if present) and only a minor part of the background.

All the patches are then resized to \(64 \times 64\) pixels. Supposing the head in a central position in the patch, even if we include minor parts of the background, we maintain only foreground, setting to 0 all the depth values greater than \(D + l\), where \(l\) is the general amount of space for a head and \(D\) is the same value computed above. Final patch values are then normalized to obtain values between \([-1, 1]\). This normalization is also required by the specific activation function that is adopted in the network architecture (see Section 3.3) and it is a fundamental step to improve CNN performance, as described in [11].
3.3. Network Architecture

We adopt a shallow network architecture, to deal with computation time and system performance. Another relevant element is the lack of publicly available annotated depth data, that force us to adopt deep models with a limited number of internal parameters. The network takes as input depth images of $64 \times 64$ pixels. There are 5 convolutional layers, where the first four have 32 filter with size of $5 \times 5$, $4 \times 4$ and $3 \times 3$ respectively, and the last one has 128 filters, with size of $3 \times 3$. Max-pooling is conducted only on the first three convolutional layers, due to the limited size of input images. Three fully connected layers are added with 128, 84 and 2 neurons respectively. We adopt the hyperbolic tangent ($tanh$) as activation function in all layers:

$$tanh(x) = \frac{2}{1 + e^{-2x}} - 1$$  \hspace{1cm} (3)

in this way network is able to map output $[-\infty, +\infty] \rightarrow [-1, +1]$. In the last fully connected layer we adopt the softmax activation, to perform the classification task.

We exploit Adam solver \[10\], with an initial learning rate set to $10^{-4}$, to resolve back-propagation and automatically adjust the learning rate values during the training phase. We exploit data augmentation technique to avoid over-fitting phenomena and increase the number of training data: each input image is flipped, so the final number of input images is doubled. As loss function, we use the categorical cross-entropy.

4. Results

In this section, experimental results of the proposed method are presented. In order to evaluate its performance, we use two public datasets, one for the patch extraction and the network training part and the second dataset for the testing phase.

Generally, head detection in depth images task lacks of the availability of publicly datasets, specifically created for face or head detection in wild contexts. Several datasets containing depth data and visible human heads were collected in this decade \([7, 2, 1]\), but they present some issues, for example they are not deep learning oriented, due to their very limited number of annotated samples. Moreover, often subjects are still, perform too static actions, and frontal face the acquisition device. Besides, we consider only dataset with ToF data, that contains frames with higher quality and depth measures accuracy as described in \[17\], in respect with structured light sensors.

We exploit Pandora dataset to generate patches of head and non-head, based on skeleton annotations, to train our CNN. Non-head candidates are extracted randomly sampling depth frames, excluding head areas. An example of extracted heads and non-heads used for the training phase is reported in Figure 3. Due to Pandora dataset features, heads with extreme poses and garments can be present.

4.1. Pandora dataset

Pandora dataset is introduced in \[3\]. It is acquired with Microsoft Kinect One device and is specifically created for the head pose estimation task. It is deep oriented, since it contains about 250k frames divided in 110 sequences of 22 subjects (10 males and 12 females). It is a challenging dataset, subjects can vary their head appearance wearing prescription glasses, sun glasses, scarves, caps and manipulate smartphones, tablets and plastic bottles that can generate head and body occlusions. It is a useful dataset to extract patches due to the presence of head with various poses and appearance. Skeleton annotations facilitate the extraction of head and non-head patches.

4.2. Cornell University dataset

Wu et al. introduced this dataset in \[21\]. It is created with the focus on modeling human activities, comprising multiple actions in a completely unsupervised setting. Like Pandora, it is collected with Microsoft Kinect One sensor for a total length of about 230 minutes, divided in 458 videos. 7 subjects perform human daily activities in 8 offices and 5 kitchens with complex backgrounds, in this way different views and head poses are guaranteed. Moreover, skeleton data are provided as ground truth annotations. Even if this dataset is not explicitly created for head detection task, it is a useful dataset to test head detection system on depth images, thanks to its variety in poses, actions and subjects.

| Methods         | True Positive | False Positive |
|------------------|---------------|----------------|
| Nghiem et al.    | 0.519         | 0.076          |
| Chen et al.      | 0.709         | 0.108          |
| Our              | **0.883**     | 0.077          |

Table 2. Results on Cornell University dataset for head detection task.
4.3. Experimental results

First, we investigate performance of the proposed system varying the size of kernel, the pixel area used to compute the average distance $D$ (see Equation 1) between a point in the scene and the acquisition camera. Kernel size affects both computation time, due to the final number of patches generated (Equation 2), and head detection rate: a bad or corrupted estimation of $D$ generate low quality patches and could compromise CNN classification performance.

In our experiments, for a fair comparison with [5], we report our results as true positives number of head detected. We consider also Intersection over Union (IoU) metric and fps value to check time performance of the system. A head is correctly detected and localized (TP, True Positive) only if

$$\text{IoU}(A, B) < \tau$$

where $A$, $B$ are ground truth and predicted head bounding boxes. According to [5], $\tau = 0.5$

If a patch is incorrectly detected as a head by CNN, it creates one false positive and one false negative. As above reported, we include also computation time, that includes the part of patch extraction and the part of CNN classification. Results are reported in [4].

As expected, system accuracy decreases and time computation increases with smaller kernel size. Kernel size can be set based on the type of final application in which head detection is necessary, where can be preferred accuracy or speed performance. Tests have been carried on a Intel i7-4790 CPU (3.60 GHz) and with a NVIDIA Quadro k2200. The deep model has been implemented and tested with Keras with Theano back-end.

Furthermore, we compare our method with two head detection systems present in literature: the first one is introduced by Chen et al. in [5]; the second one is a system for fall detection proposed in [14].

For the experimental comparison, we exploit as ground truth the skeleton data provided with the Cornell University dataset. Results are reported as the total number of True Positive and False Positive of head detection, given a subsequence of the Cornell University dataset (2785 images), the same used in [5]. This subset has been chosen by authors due to the presence of scene with good background quality, required by [14], and the presence of people, to meet the assumption present in [5] of existing heads in all input images. It is important to note that our approach do not rely on these two strong assumptions. All data with wrong ground truth annotations or missing depth data are discarded.

Table 4 reports the results. Our method achieves better performance in terms of true positive number. In particular scenes, we achieve a 100 % of correct head detections. We note that in [5, 14] is not clearly reported how the number of false positive is computed.

Figure 4. Example outputs of the proposed system. RGB frames are reported in the first row, in the second the correspondent depth ones. Our prediction is reported as blue rectangle, also ground truth (red rectangle) and other patch candidates (green) are reported. Frame are collected from Cornell University dataset.

Figure 5. Correlation between kernel size and speed performance of the proposed method. As expected, detection rate is low whith high speed performance and vice versa.

---

1By authors communication
5. Conclusions

We present a novel method to directly detect a head from depth image. Our system is based on a Convolutional Neural Network designed to classify candidates as head or non-head. Results confirm the feasibility and the accuracy of our method, that can be the first step for method about face recognition or behavior analysis, in environments that requires light invariance.

The flexibility of our approach allows the possibility of future work, that may involve the investigation of multiple head detection task in depth frame. This step requires the acquisition of new dataset, due to the lack of specific dataset for this task. Moreover, future extensions are also related with the reduction of computation time.

References

[1] A. D. Bagdanov, A. Del Bimbo, and I. Masi. The florence 2d/3d hybrid face dataset. In Proceedings of the 2011 joint ACM workshop on Human gesture and behavior understanding, pages 79–80. ACM, 2011.

[2] T. Baltrušaitis, P. Robinson, and L.-P. Morency. 3d constrained local model for rigid and non-rigid facial tracking. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pages 2610–2617. IEEE, 2012.

[3] G. Borghi, M. Venturelli, R. Vezzani, and R. Cucchiara. Poseidon: Face-from-depth for driver pose estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[4] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In Computer Vision, 2009 IEEE 12th International Conference on, pages 1365–1372. IEEE, 2009.

[5] S. Chen, F. Bremond, H. Nguyen, and H. Thomas. Exploring depth information for head detection with depth images. In Advanced Video and Signal Based Surveillance (AVSS), 2016 13th IEEE International Conference on, pages 228–234. IEEE, 2016.

[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, volume 1, pages 886–893. IEEE, 2005.

[7] G. Fanelli, T. Weise, J. Gall, and L. Van Gool. Real time head pose estimation from consumer depth cameras. In Joint Pattern Recognition Symposium, pages 101–110. Springer, 2011.

[8] S. Ikemura and H. Fujiyoshi. Real-time human detection using relational depth similarity features. Computer Vision–ACCV 2010, pages 25–38, 2011.

[9] M. H. Khan, K. Shirahama, M. S. Farid, and M. Grzegorzek. Multiple human detection in depth images. In Multimedia Signal Processing (MMSP), 2016 IEEE 18th International Workshop on, pages 1–6. IEEE, 2016.

[10] D. Kingma and J. Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pages 1097–1105, 2012.

[12] K. Levi and Y. Weiss. Learning object detection from a small number of examples: the importance of good features. In Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, volume 2, pages II–II. IEEE, 2004.

[13] D. G. Lowe. Object recognition from local scale-invariant features. In Computer vision, 1999. The proceedings of the seventh IEEE international conference on, volume 2, pages 1150–1157. Ieee, 1999.

[14] A. T. Nghiem, E. Auvinet, and J. Meunier. Head detection using kinect camera and its application to fall detection. In Information Science, Signal Processing and their Applications (ISSPA), 2012 11th International Conference on, pages 164–169. IEEE, 2012.

[15] E. Osuna, R. Freund, and F. Girosit. Training support vector machines: an application to face detection. In Computer vision and pattern recognition, 1997. Proceedings., 1997 IEEE computer society conference on, pages 130–136. IEEE, 1997.

[16] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. IEEE Transactions on pattern analysis and machine intelligence, 20(1):23–38, 1998.

[17] H. Sarbolandi, D. Lefloch, and A. Kolb. Kinect range sensing: Structured-light versus time-of-flight kinect. Computer vision and image understanding, 139:1–20, 2015.

[18] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, and R. Moore. Real-time human pose recognition in parts from single depth images. Communications of the ACM, 56(1):116–124, 2013.

[19] P. Viola and M. J. Jones. Robust real-time face detection. International journal of computer vision, 57(2):137–154, 2004.

[20] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on, volume 1, pages 90–97. IEEE, 2005.

[21] C. Wu, J. Zhang, S. Savarese, and A. Saxena. Watch-n-patch: Unsupervised understanding of actions and relations. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.

[22] L. Xia, C.-C. Chen, and J. K. Aggarwal. Human detection using depth information by kinect. In Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on, pages 15–22. IEEE, 2011.

[23] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pages 2879–2886. IEEE, 2012.