Head Tracking With Combined Face And Nose Detection

Martin Böhme, Martin Haker, Thomas Martinetz, and Erhardt Barth
Insitute for Neuro- and Bioinformatics
University of Lübeck, Germany
Email: {boehme, haker, martinetz, barth}@inb.uni-luebeck.de

Abstract— **We present a facial feature detector for time-of-flight (TOF) cameras that extends previous work by combining a nose detector based on geometric features with a face detector. The goal is to prevent false detections outside the area of the face. To detect the nose in the image, we first compute the geometric features per pixel. We then augment these geometric features with two additional features: The horizontal and vertical distance to the most likely face detected by a cascade-of-boosted-ensembles face detector. We use a very simple classifier based on an axis-aligned bounding box in feature space; pixels whose feature values fall within the box are classified as nose pixels, and all other pixels are classified as "non-nose". The extent of the bounding box is learned on a labeled training set. Despite its simiplicity, this detector already delivers satisfactory results on the geometric features alone; adding the face detector improves the equal error rate (EER) from 22.2% (without face detector) to 10.4% (with face detector). (Note when comparing with our previous results from [1] and [2] that, in contrast to this paper, the test data used there did not contain scale variations.)**

I. Introduction

In previous work, we have shown that the time-of-flight (TOF) camera, a novel type of image sensor that delivers a range map that is perfectly registered with an intensity image, can be used to implement a detector for a prominent facial feature, the nose, by combining a set of geometric features with a very simple bounding-box classifier [1]–[3].

While this simple approach already yields satisfactory results, it has its limitations, particularly if distance variations cause the apparent size of the nose in the image to change, which influences the values of the geometric features. To cope with this, the detector has to accept a larger range of feature values as "nose", and this increases the number of false detections.

One way of coping with this problem is to modify the geometric features to be scale-invariant by computing them on the reconstructed surface of the object instead of on the image [3], but this approach is not computationally efficient enough to run at camera frame rates on current hardware.

In this paper, we will examine a different approach. To improve the robustness of the detector to false detections, we will augment it with a fast face detector based on a cascade of boosted ensembles. This approach to face detection was originally described by Viola and Jones [4] and has since gained enormous popularity for a wide range of applications. As we have shown in [5], the Viola-Jones face detector, which in its original formulation operates on intensity images, can be extended to work on the combined range and intensity data of a TOF camera. This not only increases the detection performance but also reduces the running time of the face detector.

An obvious way of combining the face detector with the nose detector would be to reject any pixels that do not fall within a detected face. However, this would have the disadvantage that if the face is not detected (a false negative), the result of the nose detector is always rejected, even if it is correct. For this reason, we use a slightly different approach: For each frame, we try to find a face; if no face is found, we determine the most likely position of the face by taking the subregion of the image that generated the strongest response in the face detector. Then, for each pixel, we compute the horizontal and vertical distance to the center of the face and use these values as two additional features in addition to the geometric features. The idea is that the nose is generally close to the center of the face, i.e. pixels that are far away from the center of the face should probably not be classified as "nose".

We then apply a very simple bounding-box classifier to these features. To train this classifier, we determine the minimum and maximum feature values that are obtained for labeled nose pixels in a set of training data. In this way, we define an axis-aligned bounding box in feature space. To detect the nose in new images, we compute the feature values for each pixel; if they lie within the bounding box, the pixel is classified as "nose"; otherwise, it is classified as ”non-nose".

Once the location of the nose has been identified, it can be used to implement a head tracker. The nose has already been identified by several other authors as an important feature for head tracking [6], [7], and we have already used the previous version of our nose detector to implement a head tracker [2].

The rest of this paper is structured as follows: We will first define the geometric features and present an overview of the Viola-Jones face detector. We will then describe the simple bounding-box classifier. Finally, we will evaluate the performance of our nose detector on a database of face images and quantify the improvement in detection performance that is obtained using the face detector.

The ARTTS project is funded by the European Commission (contract no. IST-34107) within the Information Society Technologies (IST) priority of the 6th Framework Programme. This publication reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

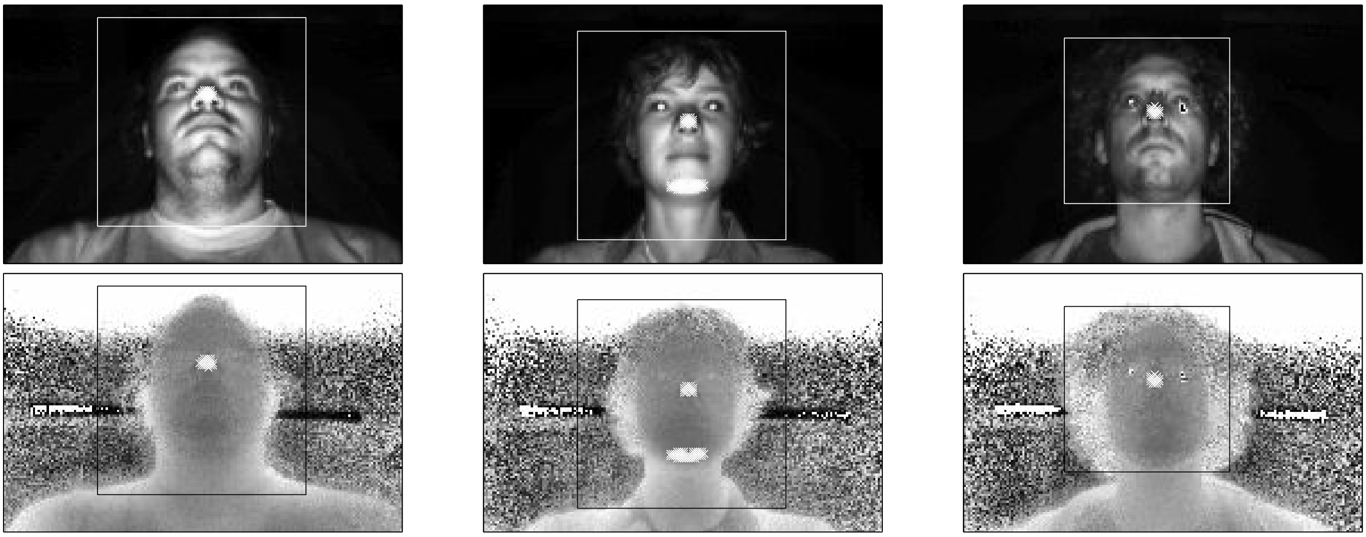

Fig. 1. Sample detection results on test images (top: intensity image, bottom: range map). The square indicates the detected face, the pixels marked in white indicate detected nose pixels. The second image shows an example of false detections (on the chin).

II. Method

A. Geometric Features

The geometric features that we use for the nose detection task, known as the *generalized eccentricities*, are related to Gaussian curvature. We will briefly review their definition here; for more detail, see [1].

We will define the generalized eccentricities on a special type of surface known as the *Monge patch* or 2-1/2-D image. Such surfaces have the property that the surface points (x, y, z) can be defined by a function f of x and y, with $z = f(x, y)$. Because the TOF camera delivers both an intensity value per pixel, we can define two such surfaces, where $f(x, y)$ is either the range or the intensity value corresponding to a pixel position (x, y). (For more details on the geometrical interpretation of images for image analysis, see [8] and [9].)

The generalized eccentricities ϵ_n, $n = 0, 1, 2, \ldots$ are now defined as

$$\epsilon_n^2 = (c_n(x,y) * l(x,y))^2 + (s_n(x,y) * l(x,y))^2 , \quad (1)$$

where c_n and s_n are filter kernels corresponding to transfer functions C_n and S_n defined in terms of polar coordinates ρ (spatial frequency) and θ (orientation):

$$\begin{aligned} C_n &= i^n A(\rho) \cos(n\theta), \\ S_n &= i^n A(\rho) \sin(n\theta), \end{aligned} \quad (2)$$

where $A(\rho)$ is a radial filter tuning function. If we set this to $A(\rho) = (2\pi\rho)^2$, the generalized eccentricities ϵ_0 and ϵ_2 can be used to distinguish between six basic surface types (see [9] for details). In practice, $A(\rho)$ is combined with a low-pass filter to reduce the sensitivity of the features to noise.

In a TOF range map, background regions can contain a relatively high amount of noise because they return little light. To avoid unwanted spatial filter responses in these regions, a threshold computed using Otsu's method [10] was applied to the intensity data to segment the foreground from the background; the background was then set to a fixed value in both the range map and the intensity image. This prevents false detections on the background; the face detector will additionally prevent false detections on the rest of the foreground, such as the person's torso.

B. Face Detector

Viola and Jones [4] introduced an approach to face detection (also known as a *cascade of boosted ensembles*) that is computationally efficient while achieving good classification performance. The Viola-Jones face detector is based on a cascade structure (see Fig. 2); early stages in the cascade require little computation but can only identify subregions in the image that are easily classified as nonfaces. These nonfaces are rejected immediately, while all other subregions (true faces and "hard" nonfaces) are passed on to the subsequent stages in the cascade. The cascade stages become progressively more sophisticated and are able to reject more and more nonfaces until all that is left (ideally) are the faces. Because most subregions in an image are very dissimilar to a face, the average number of stages that are processed per subregion and hence the average computational cost are low.

Each of the stages of the cascade consists of a classifier trained using the AdaBoost algorithm on a set of features (so-called *Haar-like* features) that can be evaluated quickly and in constant time, independent of the size of the feature, using a data structure known as an integral image.

The Viola-Jones face detector originally operates on intensity images; as we have shown in [5], the detector can be extended to use both intensity and range features on the data from a TOF camera, and this detector has better classification performance

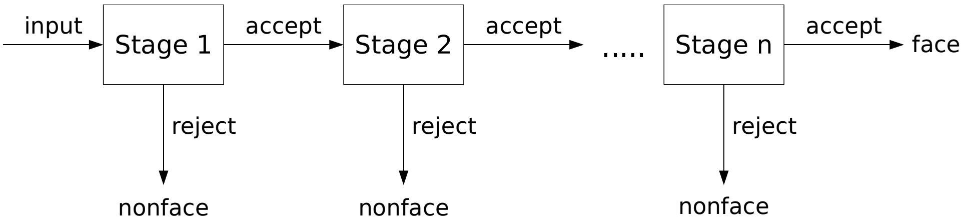

Fig. 2. Cascade structure of the Viola-Jones face detector

and lower running time than a detector operating on either type of data alone. In this paper, we will therefore use this extended face detector based on intensity and range data.

One way of integrating the face detector with the nose detector based on the geometric features would be to reject any noses that fall outside the detected face region. However, this approach has two disadvantages: (i) The nose is expected to lie near the center of the face, so if a nose is detected near the border of the face region, this is most likely a false detection that we would like to be able to identify as such. (ii) If no face is detected in the current frame (a false negative), a potentially correct nose detection is always suppressed.

For these two reasons, we use an alternative approach. We always find the most likely face subregion in the image by choosing the subregion that passed the largest number of stages in the cascade. In the case of a false negative, it is still likely that the true face passed more cascade stages than all nonface subregions. If more than one subregion passed the same maximum number of stages, we use the decision value of that stage to break ties.

When the most likely face has been found, we compute two additional features for each pixel: The horizontal and vertical distance of that pixel to the center of the detected face. Because the nose will always lie at a similar position relative to the center of the face, this will help the classifier to ignore false detections that occur far from the center of the face.

C. Bounding-Box Classifier

The classifier is based on the features ϵ_0 and ϵ_2 (see Section II-A), evaluated on both the range map and the intensity data, as well as the horizontal and vertical distance from the center of the detected face (see Section II-B). Because the feature space spanned by ϵ_0 and ϵ_2 has a radial structure (see [9]), we convert these features to polar coordinates r and ϕ before passing them on to the classifier. For each pixel, we obtain a feature vector $\mathbf{F} = (F_1, \ldots, F_6)$, which contains the values of r and ϕ for the range and intensity data as well as the horizontal and vertical distance from the center of the face.

On a set of training images, we compute the feature vectors obtained on the nose pixels (which were hand-labeled in the images). For each feature, we compute the minimum and maximum values $F_{\min j}$ and $F_{\max j}$ of that feature across all of these nose pixels. In this way, we obtain vectors $\mathbf{F}_{\min}$ and $\mathbf{F}_{\max}$ that define an axis-aligned bounding box in feature space. A pixel is classified as "nose" if its feature values fall within this bounding box and "non-nose" otherwise.

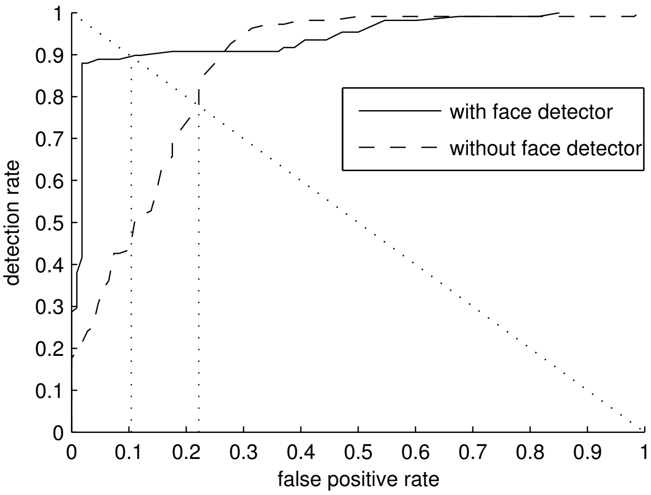

Fig. 3. ROC curves of detection rate versus false positive rate for the combined nose detector (geometric features and face detector) and for the detector based only on geometric features (no face detector). Detection rate and false positive rate are evaluated on a per-image basis, i.e. an image is counted as a detection if the nose is identified correctly and as a false positive if at least one non-nose pixel is falsely classified as "nose". Strictly speaking, therefore, the curves are not standard ROC curves, but they represent the information one is interested in for this application: How accurately does the detector give the correct response per image.

To control the tradeoff between false-positive rate and false-negative rate, we can scale the box around its center, obtaining new bounding box limits

$$\begin{aligned} \hat{\mathbf{F}}_{\min} &= \mathbf{F}_{\text{centre}} - \alpha \mathbf{F}_{\text{halfwidth}}, \\ \hat{\mathbf{F}}_{\max} &= \mathbf{F}_{\text{centre}} + \alpha \mathbf{F}_{\text{halfwidth}}, \end{aligned} \qquad (3)$$

where $\mathbf{F}_{\text{centre}} = \frac{\mathbf{F}_{\min} + \mathbf{F}_{\max}}{2}$, $\mathbf{F}_{\text{halfwidth}} = \frac{\mathbf{F}_{\max} - \mathbf{F}_{\min}}{2}$, and α is the parameter that controls the scaling of the box.

We use this simple classifier because it can be evaluated quickly and still yields relatively good results. We would expect a more sophisticated classifier, such as an SVM, to be superior; however, we are mainly interested in the effect of the face detector on classification performance, and this effect should remain fundamentally the same independent of the type of classifier that is used.

III. Results

The nose detector was evaluated on a database of images that were acquired with a MESA SR3000 TOF camera [11]. The images showed the head and upper torso of nine subjects; head pose varied between images, and the images were taken with the subjects at two different distances from the camera (60 cm and 90 cm).

Because the face detector requires a much larger set of training images, it was trained beforehand on a separate set of 5412 faces and 3486 nonface images [5]. This face detector was then used to compute the face position features for the nose detector.

The bounding-box classifier for the nose detector was then trained on three of the subjects and tested on the remaining six subjects (see Fig. 1 for some examples of detections on the test set). We determined detection rate and false positive rate on a per-image basis, i.e. an image is counted as a detection if the nose is identified correctly (within five pixels from the hand-labeled position) and as a false positive if at least one non-nose pixel is falsely classified as "nose". We believe that this gives more meaningful and easily interpreted results than computing these rates per pixel. Note that, by this definition, an image can count as both a detection and a false positive.

Fig. 3 shows the ROC (receiver operating characteristic) curves for the combined classifier (using both the geometric features and the face detector) as well as for a classifier that used only the geometric features. The combined classifier achieves a markedly better classification performance, with an equal error rate (EER) of 10.4%, compared to the EER of 22.2% for the classifier that did not use the face detector. Also, note that the detection rate for the classifier without the face detector drops off quickly as the false positive rate is reduced below the EER point; in contrast, the detection rate for the combined classifier remains almost constant as the false positive rate reduces and only drops off once the false positive rate reduces below 2%. This shows how the face detection component makes the combined nose detector much more robust against false positives. The combined classifier is at a slight disadvantage for very high false positive rates, but this is a regime that is not very interesting for practical applications.

A C++ implementation of the combined detector running on a 2.66 GHz Intel Core 2 Duo requires 18.8 ms per frame, which is equivalent to just over 50 frames per second. The detector is thus easily able to run at camera frame rates on current hardware, making it suitable for interactive applications.

One possible application is for human-machine interaction; for example, we have demonstrated how a TOF-camera-based head tracker can be used for text entry [2]. Another application is in the car, where a head tracker could be used to monitor where the driver is looking and to trigger an alert if the driver's attention leaves the road for too long.

IV. Conclusion

We have shown how an existing nose detector based on geometric features can be made substantially more robust against false positives by combining it with a face detector. In this combined detector, the face detector and the geometric features play two complementary roles: The face detector achieves a rough localization of the face and makes the detector robust against false positives that occur far away from the expected position of the nose; the geometric features allow a precise localization of the nose within the face region. Despite its increased complexity, the combined detector still runs at camera frame rates on contemporary hardware.

References

[1] M. Haker, M. Böhme, T. Martinetz, and E. Barth, "Geometric invariants for facial feature tracking with 3D TOF cameras," in *Proceedings of the IEEE International Symposium on Signals, Circuits & Systems (ISSCS)*, vol. 1, Iasi, Romania, 2007, pp. 109–112.

[2] M. Böhme, M. Haker, T. Martinetz, and E. Barth, "A facial feature tracker for human-computer interaction based on 3D Time-of-Flight cameras," *International Journal of Intelligent Systems Technologies and Applications*, vol. 5, no. 3/4, pp. 264–273, 2008.

[3] M. Haker, M. Böhme, T. Martinetz, and E. Barth, "Scale-invariant range features for time-of-flight camera applications," in *CVPR 2008 Workshop on Time-of-Flight-based Computer Vision (TOF-CV)*, 2008.

[4] P. Viola and M. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[5] M. Böhme, M. Haker, K. Riemer, T. Martinetz, and E. Barth, "Face detection using a time-of-flight camera," in *Dynamic 3D Imaging – Workshop in Conjunction with DAGM*, 2009, (submitted).

[6] L. Yin and A. Basu, "Nose shape estimation and tracking for model-based coding," in *Proc. IEEE International Conference on Acoustics, Speech, Signal Processing*, May 2001, pp. 1477–1480.

[7] D. Gorodnichy, "On importance of nose for face tracking," in *Proc. IEEE Intern. Conf. on Automatic Face and Gesture Recognition (FG'2002)*, Washington, D.C., May 2002.

[8] C. Zetzsche and E. Barth, "Fundamental limits of linear filters in the visual processing of two-dimensional signals," *Vision Research*, vol. 30, pp. 1111–1117, 1990.

[9] E. Barth, T. Caelli, and C. Zetzsche, "Image encoding, labeling, and reconstruction from differential geometry," *CVGIP: Graphical Models and Image Processing*, vol. 55, no. 6, pp. 428–46, November 1993.

[10] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, January 1979.

[11] T. Oggier, B. Büttgen, F. Lustenberger, G. Becker, B. Rüegg, and A. Hodac, "SwissRanger$^{\text{TM}}$ SR3000 and first experiences based on miniaturized 3D-TOF cameras," 2006.