Adapting Hausdorff metrics to face detection systems: a scale-normalized Hausdorff distance approach

Pablo Suau

Departamento de Ciencia de la Computación e Inteligencia Artificial
Universidad de Alicante, Ap. de correos 99, 03080, Alicante (Spain)
pablo@dccia.ua.es

Abstract. Template matching face detection systems are used very often as a previous step in several biometric applications. These biometric applications, like face recognition or video surveillance systems, need the face detection step to be efficient and robust enough to achieve better results. One of many template matching face detection methods uses Hausdorff distance in order to search the part of the image more similar to a face. Although Hausdorff distance involves very accurate results and low error rates, overall robustness can be increased if we adapt it to our concrete application. In this paper we show how to adjust Hausdorff metrics to face detection systems, presenting a scale-normalized Hausdorff distance based face detection system. Experiments show that our approach can perform an accurate face detection even with complex background or varying light conditions.

1 Introduction

Several human machine interface systems and biometric applications need to be built over a reliable face detection system. Results of these applications will not be robust enough to be useful if a correct face localization is not performed; an automated system will not be able to fully recognize a facial expression, for instance, if this fundamental step is not correct and some essential information such as eyebrows position or upper face wrinkles are not properly recognized. However, although face detection significance, several complex systems avoid this step, due to the fact that their designers consider that is not so important, and manual techniques are used to know in a quite simple way where a face is in an image [11].

Recent research surveys [1] reveal the existence of several simplified facial detection systems that could be useful as a first step of more complex systems. These systems are based on *face localization* [2], that aims to determine the image position of a single face, with the assumption that the input image only includes one.

In the case of face localization, these surveys [1] indicate that the method based on template matching and Hausdorff distance is one of the most robust.

In template matching approaches, a standard face pattern (usually frontal) is predefined, and the part of the image more similar to this pattern is searched in an image, by correlation. In [3], Jesorsky et al. describe a face localization system based on template matching, using Hausdorff distance to calculate correlation. Their method allows to find a face in an image at different scales, and gives better detection results that other face detction systems ysing differente distance metrics. However, results of this proposal and related ones are strongly affected if complex backgrounds are present. Other approaches are based on previous steps before using Hausdorff metrics, like skin color segmentation [8], but using color images is not always possible.

In this work, we present an improved face detection system based on a scale-normalized Hausdorff distance and template matching. We have chosen a face localization method because we will use it as a first step of a more complex facial expression recognition system. In section 2, we define Hausdorff distance metrics and how it can be used to find objects in an image. Then, in section 3, we explain how our face detection system works, remarking changes we have made to improve it. Finally, in section 4 we show some experimental results.

2 Hausdorff distance for template matching

Hausdorff distance is a technique for comparing sets of points; for each point of a model set, it measures how near they lie from any point of another image set and vice versa. It is used in pattern recognition to determine the degree of resemblance between two objects ([4],[5]).

Hausdorff distance can be defined as the maximum distance from a point set to the nearest point in other point set. If $A = \{a_1, ..., a_m\}$ and $B = \{b_1, ..., b_n\}$ are two point sets, the Hausdorff distance between A and B is defined as

$$H(A, B) = \max(h(A, B), h(B, A)) \tag{1}$$

where

$$h(A, B) = \max_{a \epsilon A} \min_{b \epsilon B} \|a - b\| \tag{2}$$

is called the *directed Hausdorff distance* from set A to set B with some underlying norm $\| \cdot \|$ on the points of A and B, and

$$h(B, A) = \max_{b \epsilon B} \min_{a \epsilon A} \|b - a\| \tag{3}$$

is called the *reverse Hausdorff distance.* As we can see, Haussdorff distance is not symmetric. It gives an interesting measure of the mutual proximity of two set of points, by indicating the maximal distance between any point of one set to other set.

Two main problems exist when we use Hausdorff distance for image processing: scale (which will be discussed in section 3) and outlier points. The experiments in [7] have proven that outlier points effect can be reduced using an average version of the Hausdorff distance, based on taking the average of the single point distances. In fact, this Hausdorff variant gives better results than other known Hausdorff metrics ([7]). The average Hausdorff distance is defined as:

$$h_{avg}(A, B) = \frac{1}{|A|} \sum_{a \epsilon A} \min_{b \epsilon B} \|a - b\| \tag{4}$$

$$h_{avg}(B, A) = \frac{1}{|B|} \sum_{b \epsilon B} \min_{a \epsilon A} \|b - a\| \tag{5}$$

As we have said before, Hausdorff distance is very useful to detect a single object in an image. Let A be a representation of the image where we want to look for an object, and B a representation of the object itself. Let also $T = (t_x, t_y, s_x, s_y)$ be our transformation space, where t_x and t_y represent translation, and s_x and s_y scale, and P a set containing all possible values for T (so, if $b = (b_x, b_y)$ is a point of B and $p \epsilon P$, then $T_p(b) = (s_x \cdot b_x + t_x, s_y \cdot b_y + t_y)$. Allowed transformations and their parameter space P depend on the application). Figure 1 shows the process to find a model B in an image A using transformations.

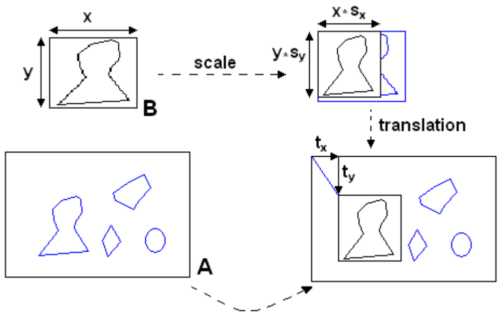

Fig. 1. Applying transformations to detect a model in an image.

In order to find B inside A, we need to search for transformation parameters values $p \epsilon P$ such that the Hausdorff distance between the transformed model $T_p(B)$ and A is minimized. The Hausdorff distance to be minimized is:

$$d_{\hat{p}} = \min_{p \epsilon P} H(A, T_p(B)) \tag{6}$$

where $\hat{p}$ is the set of values $p \epsilon P$ for the transformation space parameters T that minimize the Hausdorff distance between the image A and the transformed model B.

So, $f_B(T_p) = h(T_p(B), A)$ is called the *forward distance* and will be used to find the object in the image, and $f_A(T_p) = h(A, T_p(B))$ is called the *reverse distance* and will allow the system to detect a false positive. We must note that only the part of the image that is covered by the model must be considered.

3 System description

We expect to use our face detection approach as part of a more complex facial expression recognition system. Therefore, as in [3], our method is based on face localization: only one face is present in test images at any time, and the part of the image more similar to a face is searched. As a consequence, we can have a complete system including face detection, and focus more on our main goal.

Figure 2 shows the complete face localization process, which is divided in a segmentation step and a localization step. The segmentation step transforms the input image into a binary edge image, so a template matching search can be performed on it. Once the segmentation is finished, the localization step uses Eq.(6) as a distance metrics to compute a template matching algorithm using a face template, in order to detect the exact position and size of the face in the image. These two steps are described in detail in the following sections.

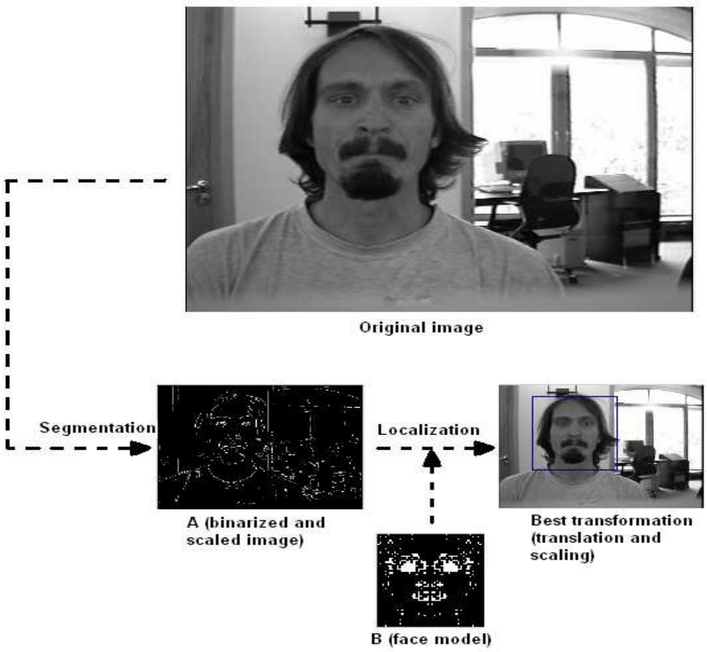

Fig. 2. A summary of the face localization system.

3.1 Segmentation phase

The input image must be transformed into a binary edge image prior to starting the localization step. This binary image should contain enough information to get correct localization results.

To extract edges from the input image, we could use several edge operators, like Canny, Sobel, and so on, being the Sobel operator the most used in template matching systems combined with Hausdorff distance metrics ([3], [4], [5]). However, as we can see in Figure 3a and Figure 3b, using a simple Sobel operator in an image with complex background can be inappropriate. Sobel edge detection with a low threshold (Figure 3a) results in an image containing too much edges, including a high amount of background points that can affect the localization step. Sobel edge detection with a higher threshold (Figure 3b) slightly decreases the number of edges detected, but it also makes points related to facial features like eyes and mouth disappear; furthermore, points from complex background still are present. Our system solves this problem combining two Sobel edge detections with different thresholds (Figure 3c). We calculate the difference between the edge image obtained after a Sobel edge detection with a low threshold and another obtained using a high threshold. This operation makes most of background points disappear, while points obtained from facial features segmentation still remain.

(a) (b) (c)

Fig. 3. a) Sobel edge detection using threshold = 0.04. b) Sobel edge detection using threshold = 0.09. c) Edge detection combining multiple Sobel operators.

In order to increase the system speed, and knowing the fact that the Hausdorff distance computing time depends on the number of points of sets A and B, the input image is scaled *before* the image segmentation process starts. Our experimental results show that the input image could be scaled up to a 40% of its original size without affecting the final outcome and decreasing considerably the computation time.

3.2 Localization phase

After the segmentation step, we obtain a scaled binary image where some patterns of points representing a face could be found. The template we will use is the same as in [3] (Figure 4). As shown in [6], this point template is created from genetic algorithms as an average of a set of faces. Almost all points represent facial features like eyes, mouth and nose, while some other points represent face boundaries.

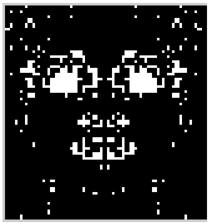

Fig. 4. Face template.

Using the face model B and the segmented binary image A, we try to find the face in the image using Eq.(6), testing different scales and translations for B. The values for the transformation set $\hat{p}$ that minimizes $H(A, T_p(B))$ gives us the position and size of the face in the image.

As discussed before, the scale is one of the main practical problems of Hausdorff distance when it is used for image processing. Due to the fact that the size of faces could be highly variable, we must try a high range of scale values (s_x, s_y) for the model B during the search. The problem is that Hausdorff distance values are lower for smaller template sizes; which means that if the face in the image is large, although correct template transformations for scale and translation results in a low Hausdorff distance, the smallest values of scale applied to the template generate even smaller distance values. The consequences can be seen in Figure 5a. It seems that a normalization factor is needed to adapt the Hausdorff distance metrics to different template sizes. It must be chosen carefully, since modifying the distance substantially can result in the opposite effect: smaller faces would be very difficult to detect.

After completing the experimentation process (that will be discussed in next section), we found that dividing $h(T_p(B), A)$, and $h(A, T_p(B))$ by $\sqrt{\frac{s_x+s_y}{2}}$, allows the system to detect exactly the size and position of faces, without any scale effect. Figure 5b shows how this normalization factor solves the scale issue.

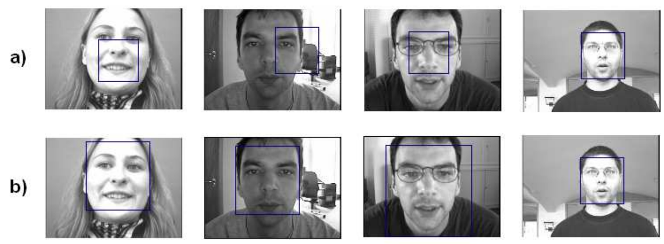

Fig. 5. Scale issue. a) face detection without normalization factor. b) face detection with normalization factor. Normalization factor is flexible enough to allow detection of any size faces.

A correct combination of the forward and reverse Hausdorff distance can be very useful to avoid false positives. In [3], authors use the product of the two distances to help in a face/non-face decision. Figure 6a shows some examples of incorrect face localization when only forward distance is used. The point is to find the transformation set $\hat{p}$ that minimizes the product of the forward distance $h(A, T_{\hat{p}}(B))$ and the reverse distance $h(T_{\hat{p}}(B), A)$. Figure 6b shows face localization results for the same images but using the reverse distance to discard false positives.

a)

b)

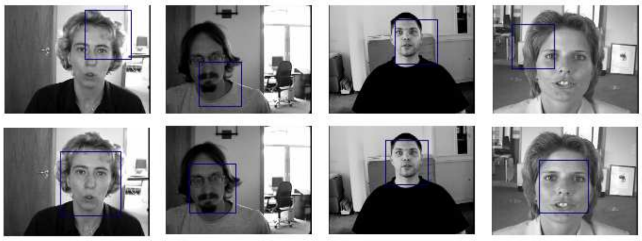

Fig. 6. Using the reverse distance to discard false positives. a) Face detection without reverse distance. b) Face detection with reverse distance. Reverse distance, in combination with forward distance, helps in discarding parts in an image that are not faces.

4 Experimental results

This section is divided in two subsections. In section 4.1 we will demonstrate that our normalization factor improves the Hausdorff metrics results for face detection. In section 4.2 we will show our system results.

The test set used in experiments was the BioID database[1], that is available for free on the Internet and includes 1521 gray level images, having all of them a size of 384x288 pixels. This complete database contains images of 23 different people, having a wide variety of light conditions, face sizes, and complex backgrounds. The binary template associated to this database is shown in Figure 4 and its size is 81x86 pixels.

The search region is restricted to a centered square in the image, covering the whole image height. Before the process, we scale the input image to a 40% of its original size. In the segmentation phase, the two edge images created from the two different Sobel operators (with thresholds 0.04 and 0.09, respectively) are subtracted. The scaling parameter range for the template is between 55% and 120%.

[1] http://www.humanscan.de/support/downloads/facedb.php

4.1 Normalization factor

Face localization based on Hausdorff distance needs a template to compare with. This template is scaled so faces with different sizes can be detected. To demonstrate that distance and template scale are related, we calculated, for every image in the BioID database, the minimum distance obtained for every template scale. As we can see in Figure 7a, distance increases in a quadratic way with template scale, so bigger faces (Figure 5a) are not correctly detected because although correct template transformations for scale and translation result in a low distance, the smallest values of scale applied to the template generate even smaller values.

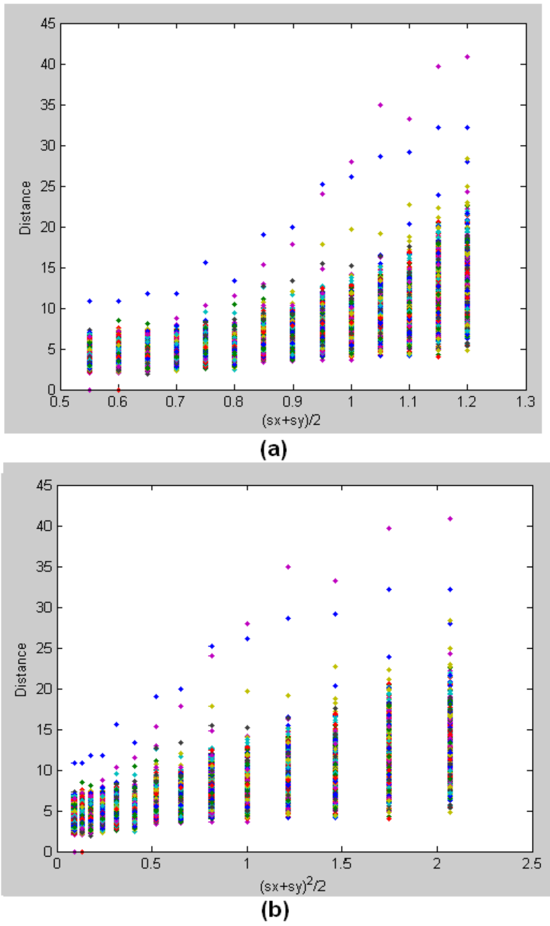

Fig. 7. Correlation between Hausdorff distance and template scale. a) Distance versus scale. b) Distance versus $scale^2$.

A more formal demonstration can be performed using the Pearson's correlation coefficient, that is a measure of the strength of the association between two variables, used quite often in statistics. Pearson's correlation coefficient value ranges from -1 to +1, being this value higher when a strong relationship exists. The Pearson's correlation coefficient value for our two variables (Hausdorff distance and template scale) is $p = 0.795$ with a significance level of 99%; it means that there is a strong lineal relationship.

However, Figure 7a shows that relationship between these two variables seems more quadratic than linear. When a correlation is strong but clearly non linear, it is common to transform one or both of the variables in order to remove the curvilinearity and then recalculate the correlation. In our case we created a graph where distance versus $[\frac{s_x+s_y}{2}]^2$ is shown (Figure 7b). After this transformation, the Pearson's correlation coefficient value is $p = 0.8278$ with a significance level of 99%. Therefore, quadratic relationship between these two variables is stronger than lineal relationship.

Thus, we must be aware of this quadratic dependance when we define the normalization factor. If we use $\frac{s_x+s_y}{2}$ as normalization factor, distance will be substantially modified, and it will result in the opposite effect: smaller faces will be quite difficult to detect. However, using $\sqrt{\frac{s_x+s_y}{2}}$ as normalization factor reduces the distance value significantly for larger scale values, and more slightly for smaller templates, resulting in a lower error rate for face localization (Figure 5b).

4.2 Face localization results

If we draw a bounding box where the face has been detected, we consider that a face is correctly detected when it contains all important facial features (eyes, nose, mouth, etc.) and when the face fills at least 80% of this bounding box. The error rate of our system was 7.3%, using BioID database and $\sqrt{\frac{s_x+s_y}{2}}$ as normalization factor. In the case that we use $\frac{s_x+s_y}{2}$ as normalization factor, the error rate is increased to 23.73%, due to the fact that Hausdorff distance is strongly affected for low scale values.

Other systems using BioID database to test its results have a higher error rate [9], or use less restrictive conditions to indicate when a face has been correctly detected [10]. In Figure 8a we can see some correct detections, including different scales, poses and light conditions. This fact demonstrates that our system is flexible enough to locate properly some out of plane rotated faces, even using a standard frontal face template. Our segmentation method helps in finding faces in images where a complex background exists, with varying light conditions, as we can see in Figure 8a, as well.

In Figure 8 we can see also how the normalization factor affects the face localization. Figure 8a shows correct face localization using the $\sqrt{\frac{s_x+s_y}{2}}$ as normalization factor. In Figure 8b the results for the same images using $\frac{s_x+s_y}{2}$ as normalization factor can be seen. Finally, Figure 8c shows the results for the same images, but without using any normalization factor. Thus, Figure 8 illustrates how large faces are correctly detected with our normalization factor, without affecting small faces localization.

Fig. 8. a) Some examples of correct face detection, including not frontal faces and complex backgrounds, using $\sqrt{\frac{s_x+s_y}{2}}$ as normalization factor. b) Results for the same images using $\frac{s_x+s_y}{2}$ as normalization factor. c) Results for the same images without any normalization factor.

5 Conclusions and future work

We have presented a face detection system based on template matching in an edge image using a scale-normalized Hausdorff distance. Results are better in robustness than in previous similar works. Our system is flexible enough to detect faces in images with complex backgrounds under varying light conditions, and can detect some not frontal faces.

We have introduced a normalization factor in the Hausdorff distance calculation, so this metric is not affected by template size. We have also proven that is possible to highly reduce the input image's size without affecting facial detection and resulting in a much faster process. Our segmentation method allows not interesting background points to be removed.

Future work could focus on study how some Hausdorff distance improvements can affect our detection results. It could be interesting to test if there is any gain in using a multiresolution Hausdorff distance. It could be also interesting to check if it is possible to combine several templates to try to find faces in different orientations. Finally, our face detection system would be used as a part of an automated facial expression recognition system, which is currently being developed.

References

[1] Yang, M. H., Kriegman, D., Ahuja, N., Detecting Faces in Images: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 24, no. 1, pp. 34-58, 2002.

[2] Lam, K., Yan, H., Fast Algorithm for Locating Head Boundaries, *J. Electronic Imaging*, vol.3, no.4, pp. 351-359, 1994.

[3] Jesorsky, O., Kirchberg, K. J., Frischholz, R. W., Robust Face Detection Using the Hausdorff Distance, *Third International Conference on Audio and Video-based Biometric Person Authentication*, Springer Lecture Notes in Computer Science, LNCS-2091, pp. 90-95, Halmstad, Sweden, 2001.

[4] Huttenlocher, D. P., Klanderman, G. A., Rucklidge, W. J., Comparing Images Using the Hausdorff Distance, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 9, pp. 850-853, 1993.

[5] Huttenlocher, D. P., Rucklidge, W. J., A multi-resolution technique for comparing images using the Hausdorff distance, *Technical Report 1321, Cornell University, Department of Computer Science*, 1992.

[6] Kirchberg, K. J., Jesorsky, O., Frischholz, R. W., Genetic Model Optimization for Hausdorff Distance-Based Face Localization, *International ECCV 2002 Workshop on Biometric Authentication*, Springer Lecture Notes in Computer Science, LNCS-2359, pp. 103-111, Copenhagen, Denmark, 2002.

[7] Shapiro, M. D., Blaschko, M. B., On Hausdorff Distance Measures, *Technical Report UM-CS-2004-071, Department of Computer Science, University of Massachusetts Amherst*, 2004.

[8] S. Srisuk, W. Kurutach and K. Limpitikeat, A Novel Approach for Robust, Fast and Accurate Face Detection, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems (IJUFKS)*, Vol. 9, No. 6, pp. 769-779, December 2001, World Scientific Publishing Company.

[9] V. Manian, A. Ross, A Texture-based Approach to Face Detection, *Biometric Consortium Conference (BCC)*, (Crystal City, VA), September 2004.

[10] B. Frba, C. Kblbeck, Robust Face Detection at Video Frame Rate Based on Edge Orientation Features, *Fifth IEEE International Conference on Automatic Face and Gesture Recognition (FGR 2002)*, pp. 342-347, Washington, USA, 2002.

[11] M. Rosenblum, Y. Yacoob, L. Davis, Human Expression Recognition from Motion using a Radial Basis Function Network Architecture, *IEEE Transactions on Neural Networks*, Vol. 7, no. 5, pp. 1121-1138, 1996.