An Automatic Hybrid Segmentation Approach for Aligned Face Portrait Images

Martin Hirzer[1], Martin Urschler[1], Horst Bischof[1], Josef A. Birchbauer[2]

[1] Institute for Computer Graphics and Vision
Graz University of Technology, Austria
{hirzer,urschler,bischof}@icg.tugraz.at

[2] Siemens Biometrics Center
Siemens IT Solutions and Services, Siemens Austria
josef-alois.birchbauer@siemens.com

Abstract
With the introduction of electronic personal documents (e.g. passports) in recent years the analysis of suitable photographs has become an important field of research. Such photographs for machine readable travel documents have to fulfill a set of minimal quality requirements defined by the International Civil Aviation Organization (ICAO). As some of the specified requirements are related to certain image regions only, these regions must be located in advance. In this work we present an automatic segmentation method for aligned color face images. The method is based on a convex variational energy formulation which is solved using weighted total variation. We apply constraints in the form of prior knowledge about the spatial configuration of typical passport photographs in order to solve for a global energy minimum. Several experiments on face images from two different datasets are presented to evaluate the performance of our algorithm. The obtained results demonstrate that our method is fairly robust and significantly outperforms other methods targeted at the same problem, in particular an expert system and an AdaBoost classifier.

1 Introduction

In biometrics related applications like person verification/recognition [1, 18], video surveillance or facial expression recognition [7] the analysis of facial images plays an important role. In this work we are specifically interested in analyzing facial portrait images in the context of the International Civil Aviation Organization (ICAO) standard [9] for machine readable travel documents (MRTD). The main intention of this standard is to define how arbitrary images have to be prepared in order to perform robust and highly accurate face recognition/verification. Therefore images have to be geometrically aligned and require standardized imaging conditions and facial expressions. A part of the ICAO specification describes a standardized coordinate frame based on eye locations to incorporate geometrical alignment, see Figure 1 for a definition of this geometrical coordinate system forming aligned face portrait images [14]. Other parts deal with a definition of parameters for image quality and proper facial expression and pose that classify images as suitable or improper for documents like e.g. passports in order to perform face recognition. Examples for these parameters are a frontal pose, neutral face expression with open eyes, no objects (e.g. hair) covering a face, and a uniform background.

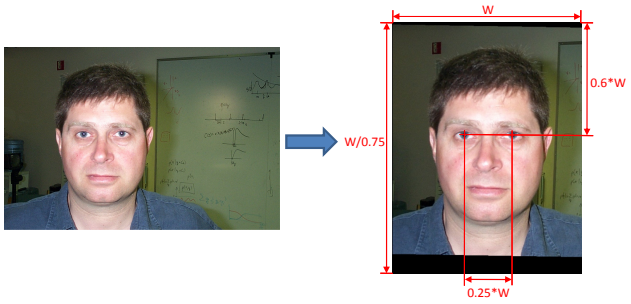


Figure 1: Illustration of the geometrical alignment procedure to form a aligned face portrait image (right) given an arbitrary input image (left). The black stripes in the right image denote a so called padding frame, i.e. an area where no corresponding data exists in the input image. The image is taken from the Caltech face database [3].

Analysis of face portrait images for preparing face recognition is significantly simplified if one is able to provide a segmentation of the image into several meaningful regions like face, hair, shoulders and background. An overview of segmentation methods including color segmentation can be found in [13]. In this work our contribution is a fully automatic hybrid region- and edge based segmentation algorithm for the separation of face portrait images into these regions. The algorithm requires aligned images according to Figure 1 as its input and provides a labeling into the four different regions. Related work on face image segmentation on aligned images is rare, one related publication is [15], where the authors show an expert system based algorithm for the task at hand.

The basic building block of our algorithm is a geodesic active contour (GAC) edge-based segmentation [4] which is formulated in a variational framework using weighted total variation [2]. This provides us with a means to perform a globally optimal separation of a foreground and a background region. As a region constraint we model two competing regions using color, texture and shape probability distributions and place these models as constraints into the globally optimal GAC framework. This way our hybrid scheme makes use of region and edge information simultaneously. Note that this algorithm is similar to the work presented in [16]. However, we do not require interaction to set up the constraints for the optimization, but instead use fore- and background probability estimates initialized from prior knowledge about the spatial location of the different regions in aligned face portrait images. Finally, we use the basic algorithm for separating two regions in a multi-region framework where different regions are able to compete against each other alternately. This is performed in a multi-resolution setup to provide efficiency and more robustness by incorporating our spatial constraints.

In the following we derive our novel face segmentation algorithm in Section 2. Section 3 describes our experiments performed on a large database of face portrait images with manually annotated ground truth in the form of segmentation labels. We compare our method against the algorithm of [15] and against a method based on an AdaBoost classification of the image pixels using a MeanShift over-segmentation as its features [6]. We are able to significantly outperform both of these methods. Finally, we conclude our work in Section 4.

2 Face Segmentation Algorithm

To solve the task of segmenting color passport photographs we use an automatic knowledge-based approach. We segment images by letting well defined start regions grow using gradient, color, texture

and shape information. The basic steps of the proposed method are image pre-processing, definition of start regions, region growing and post-processing. Prior knowledge about typical passport photographs is incorporated into the segmentation algorithm whenever possible, in particular in the definition of start regions, the region growing process and the post-processing stage.

2.1 Image Pre-Processing

First, the input image is transformed from the RGB to the Lab color space due to the drawbacks of the RGB space for color analysis. As outlined by Vezhnevets et al. in [17], the RGB space suffers from highly correlated color channels, significant perceptual non-uniformity and mixing of luminance and chrominance data. After that, a median filter is applied to the image. It reduces image noise within regions, but preserves borders at the same time.

2.2 Region Growing

As already mentioned, the basic idea is to define a start region and then extend it to image areas with similar color and texture properties, respectively. In addition the image gradient and knowledge about the shape of the expected regions are incorporated into the growing process. The growing process is based on the following four features: color, texture, gradient and shape.

2.2.1 The $\mathrm{TV_g}$-Region Model

For our hybrid growing algorithm we combine a geodesic active contour [4], which takes image gradients by means of some general edge detector function g into account, with a region model that incorporates color, texture and shape information. We developed the following energy minimization problem, where C is the evolving contour, i.e. the boundary of the growing region, $L(C)$ is its Euclidean length, u is a characteristic function, and f is a function that encompasses the region information:

$$\min_C \Big\{ E_{GAC\,Region} \Big\} = \min_C \Big\{ \underbrace{\int_0^{L(C)} g(|\nabla I(C(s))|)\, ds}_{\text{GAC}} + \lambda \underbrace{\int_\Omega u f \, d\Omega}_{\text{Region}} \Big\} \tag{1}$$

However, due to its non-convexity we replace the GAC-term in this equation with the weighted total variation $\mathrm{TV_g}$, which was introduced by Bresson et al. in [2]:

$$\min_u \Big\{ E_{TV_g\,Region} \Big\} = \min_u \Big\{ \underbrace{\int_\Omega g\,|\nabla u|\; d\Omega}_{\mathrm{TV_g}} + \lambda \underbrace{\int_\Omega u f \, d\Omega}_{\text{Region}} \Big\} \tag{2}$$

In [2] Bresson et al. proved that if u is a characteristic function 1_{Ω_C} of a set Ω_C whose boundary is denoted C, and u is allowed to vary continuously between $[0, 1]$, Equation (1) and Equation (2) describe the same energy. The advantage of the latter formulation over the previous one is its convexity, making it possible to derive a globally optimal solution. For the weighting g we use $g(|\nabla I|) = e^{-\eta|\nabla I|^\kappa}$, an edge measure that is optimized for natural images [8]. With the parameter λ the influence of the $\mathrm{TV_g}$-term and region-term on the segmentation result can be controlled. A small λ means that the result is primarily determined by the $\mathrm{TV_g}$-term, whereas a large λ makes the region-term to the main contributor. We define the function f as a probability ratio that forces the segmentation to partition

the image into homogeneous regions. The involved probabilities encompass color, texture and shape information for the corresponding region:

$$\min_u \Big\{ E_{TV_g\ Region} \Big\} = \min_u \Big\{ \underbrace{\int_\Omega g\,|\nabla u|\ d\Omega}_{\text{TV}_\text{g}} + \lambda \underbrace{\int_\Omega u \log \frac{p_2}{p_1}\, d\Omega}_{\text{Region}} \Big\} \tag{3}$$

The probability p_1 in Equation (3) always represents a foreground region, i.e. the current segmentation region under investigation. Probability p_2 represents the rest of the image competing with the foreground, which we denote as background region of the growing process. Note that in this context background does not refer to the semantical background of a face portrait image, but solely the antagonist of a currently growing region.

2.2.2 Solving the TV_g-Region Model

Solving total variation models is a demanding task due to the non-differentiability of the L_1-norm in the TV-term at zero. Many different approaches exist, ranging from explicit time marching algorithms to graph cut methods. A short overview with corresponding references can be found in [12]. We use an iterative method proposed by Chambolle et al. in [5]. It is based on introducing a second variable v and leads to the following convex energy optimization problem:

$$\min_{u,v} \Big\{ E_{TV_g\ Region} \Big\} = \min_{u,v} \Big\{ \int_\Omega g \Big| \nabla u \Big|\, d\Omega + \frac{1}{2\theta} \int_\Omega (u-v)^2\, d\Omega + \lambda \int_\Omega v \log \frac{p_2}{p_1}\, d\Omega \Big\} \tag{4}$$

Introducing a second variable v leads to a third term in the TV_g-region model, the connection term $\frac{1}{2\theta}\int_\Omega (u-v)^2\, d\Omega$. The parameter θ controls the influence of the TV_g-term representing edges and the region-term, respectively. The minimization task is now split into two steps. First, the energy is minimized in terms of u with v being fixed, and then the energy is minimized in terms of v with u being fixed. These two steps are iterated until convergence:

$$1. \qquad \min_u \Big\{ \int_\Omega g\,|\nabla u|\, d\Omega + \frac{1}{2\theta} \int_\Omega (u-v)^2\, d\Omega \Big\} \tag{5}$$

$$2. \qquad \min_v \Big\{ \frac{1}{2\theta} \int_\Omega (u-v)^2\, d\Omega + \lambda \int_\Omega v \log \frac{p_2}{p_1}\, d\Omega \Big\} \tag{6}$$

According to Chambolle et al. Equation (5) can be solved by using a dual variable $\boldsymbol{p} = \frac{\nabla u}{|\nabla u|}$:

$$\begin{aligned} v = \text{constant} \qquad & u^{n+1} = v + \theta \operatorname{div} \boldsymbol{p} \\ & \boldsymbol{p}^{n+1} = \frac{\boldsymbol{p}^n + \frac{\tau}{\theta}\nabla u}{1 + \frac{\tau}{\theta}\frac{|\nabla u|}{g}} \end{aligned} \tag{7}$$

In practice the timestep τ has to be less or equal than $\frac{1}{4}$ in order to achieve convergence. For Equation (6) we obtain:

$$\begin{aligned} u = \text{constant} \qquad & \frac{1}{\theta}(v-u) + \lambda \log \frac{p_2}{p_1} \stackrel{!}{=} 0 \\ & \Rightarrow v = u - \lambda\theta \log \frac{p_2}{p_1}, \qquad v = \max(0, \min(1, v)) \end{aligned} \tag{8}$$

After every iteration of the region growing process (Equations (7) and (8)) the probabilities p_1 and p_2 can be updated according to the actual segmentation. However, to decrease the runtime of our algorithm new probability values are only calculated after significant changes in the region topology. Furthermore we also keep track of the variation of u and v for all pixels in the image. Pixels where the variation of these variables is very low are considered to be steady, so they are excluded from further iterations. In this way we gain a notable speedup, especially for larger images.

Equation (3) just allows two regions to compete against each other, but we have to segment up to four different regions. The easiest way of solving this problem is to let the individual regions grow successively. But this leads to a dependency of the result on the growing order. To avoid that one of the regions is preferred, the regions grow alternately. We start with the first region and perform one iteration of the growing process, then we move on to the next region and again perform one iteration. This process continues until all regions have completed the first iteration. After that, the growing algorithm starts the second iteration on all regions, and so on. In this way we can let multiple regions grow virtually simultaneously.

2.2.3 Probability Calculation

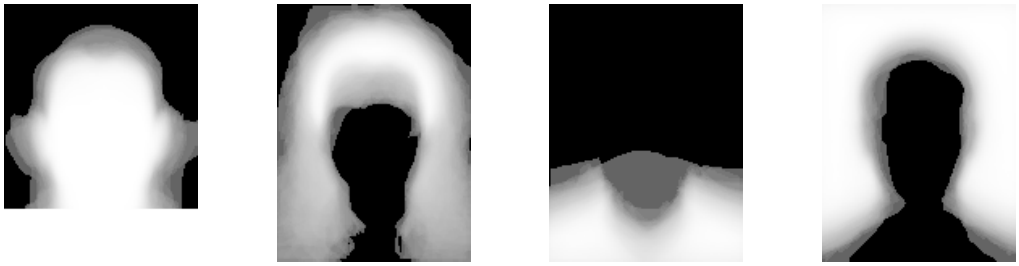
Region Probability p_1 To obtain the probability p_1 in Equation (3), we build a Gaussian model in the feature space. The feature space is either three dimensional (three color channels) or four dimensional (three color channels and one texture channel), depending on the growing region (the texture channel is built by calculating the local standard deviation of the gradient image and is only used for the hair region). The model parameters are derived from all pixels that are currently inside the region. Having built the Gaussian model for a certain region then enables us to calculate the probability that a pixel belongs to this region based on its color (and texture).

Background Probability p_2 To derive the probability p_2 in case of several simultaneously growing regions, we use their Gaussian models. The calculation is almost the same as in case of the growing region's probability term p_1. The only difference is that we now have to combine multiple Gaussian models. For each pixel, p_2 is the maximum of the probability values derived from the models of the current background regions, i.e. all regions except the currently growing one. In this way the growing region encounters maximum resistance and will only extend to image areas that have a smaller probability for all other regions. Note that there are no additional computational costs involved in building the Gaussian models for calculating p_2, because usually, if multiple regions are present in the image, we let all of them grow simultaneously. This means that we need these models anyway for calculation of the regions' probabilities p_1. Furthermore we also have to deal with the case of only one growing region. For example, when extending the face start region, we do not have any knowledge about the other regions. Hence we can not define a competing region for the growing face region. Instead we simply define a fixed threshold for p_2. With this value we can adjust how easy or hard it is for the region to grow. The higher the value the harder it is for the region to extend.

Multimodal Probability Calculation For the face, hair and shoulder region a single Gaussian model in the feature space is usually sufficient to represent the region properties well. However, the situation is different for the background region. While a uniform background region can of course also be represented by a single Gaussian model, such a model is likely to fail in case of a non-uniform

background region. To avoid this problem, we use multiple Gaussian models for the background region. Using the MeanShift segmentation algorithm [13] we identify all different subregions within the background region. For each of these subregions a Gaussian model is derived. The final probability p_1 for every pixel is obtained in the same way as the probability values p_2 when multiple regions are growing simultaneously. It is defined as the maximum among all probability values derived from the subregion models.

Incorporation of Shape Information The last step of the probability calculation is the incorporation of shape information. This is achieved by weighting the calculated probabilities p_1 with shape probability maps, as shown in Figure 2. Since the probability values for p_2 are derived from the values of p_1 (except for the case when a fixed threshold is used), the shape information is also included there. In Figure 3 the effect of weighting the probabilities is presented. One can see how a region's probability is diminished in improper image areas after multiplication with the corresponding shape probability map.



(a) Face (b) Hair (c) Shoulder (d) Background

Figure 2: Shape probability maps calculated from 439 hand labeled face images. The higher the probability of a pixel is the whiter it appears in the image.

(a) Image (b) Shoulder probability (c) Weighted shoulder probability

Figure 3: Incorporation of shape information into the probability calculation. The image is taken from the Caltech face database [3].

2.3 Work Flow Description

Before segmenting a certain image region an initial start region has to be defined, which serves as a starting point for the region growing process. The only constraint on the input face images is that they have to be aligned with respect to eye position. The only assumption that we can make at the beginning of the segmentation process is that there is a face located in the middle of the image.

For efficiency reasons we use a multi-resolution approach. First, the geometrical resolution of the input image is reduced and the segmentation process is applied to the smaller image. This first segmentation part consists of two steps, the first and the second segmentation stage, both carried out on the low resolution image. The purpose of the first segmentation stage is a rough localization and segmentation of individual image regions. Starting from the face region the algorithm tries to find a hair, a shoulder and finally a background region. Since we want to avoid that regions grow into unsuitable image areas during this first stage, we use rather strict parameter settings. After finishing the first stage, we start a second segmentation stage to enhance the first segmentation result. This time we use relaxed parameters and let all regions found in the previous stage grow simultaneously. Finally, the result is scaled up to the original image resolution again and serves as starting point for a final segmentation stage. Due to the good initialization, this final stage is very efficient despite the full image resolution.

2.3.1 First Segmentation Stage

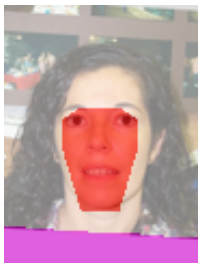
Background Uniformity Test Before actually starting with the face region we try to estimate whether the background is uniform or not. We define a test region located in the upper image where the background is usually located. Then the standard deviation of the pixels' color values within the test region is examined. If it is small enough, the background region is considered to be uniform, and we can easily segment it by letting the test region grow.

Face Region In this step the face region is segmented. Figure 4 shows the corresponding start region located in the image center. To increase the performance of our algorithm, we use a skin detector based on two statistical Gaussian mixture models in color space [10]. One model represents the skin class, and the other the non-skin class. Using Equation (9) with probabilities derived from these two models one can classify a pixel as skin or non-skin. The classification result depends on a specified threshold T. The higher this threshold is the less skin pixels will be detected:

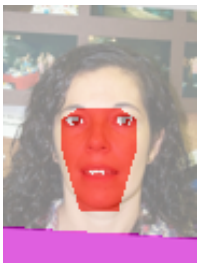
$$\frac{P(\boldsymbol{c}|skin)}{P(\boldsymbol{c}|nonSkin)} \geqslant T \tag{9}$$

Our algorithm now applies the skin detector to the face start region. Pixels that are labeled as non-skin are removed from the start region. In doing so we gain robustness in cases where parts of the face are covered, e.g. by glasses or a beard. Having defined the start region properly, we can now let it grow in order to find the face region.

Hair and Shoulder Region The next stage is the definition of start regions for the hair and shoulder region. We know that hair is usually located above the face. Special cases are bald and half-bald people, which have no hair or only little hair near the ears respectively. The position of the shoulder regions is underneath the head, in the left and right image corner. To find suitable start regions, we use the already segmented face region. First, we calculate the convex hull of the face region. Note that there can be some outlier face pixels located around the face region. To avoid distortions caused by these pixels, we use only the largest part of the face region for calculating the convex hull. The portion of the convex hull that lies above the eyes and is not already covered by another region (e.g. uniform background region) is then used as start region for the hair region. To define the shoulder



(a) Entire face start region



(b) Skin inside face start region

(c) Skin inside face start region

Figure 4: The face start region is located in the image center (red). Applying the skin classifier gives us the final start region. Purple regions mark padding frames (known), the green region in Figure 4c denotes an already segmented, uniform background region. The images are taken from the Caltech face database [3] and the FERET database [11].

start region, we use four points: the lower left image corner, the lower right image corner, and two points that are located approximately at the left and right boundary of a person's neck. This results in a trapezoid start region. Again, only the part that is not already covered by another region is actually used. Both start regions are shown in Figure 5a.

The next step depends on the background uniformity estimated at the beginning of the segmentation procedure. If the background seems to be uniform, we directly jump to the second segmentation stage (Section 2.3.2), which means that all regions currently present in the image grow simultaneously with relaxed parameters. The fact that the hair and shoulder region are only represented by their start regions is negligible, because the objective of the first segmentation stage, a rough localization and segmentation of the image, has already been achieved at this point. However, if the background uniformity test indicated a non-uniform background region, the background has not been segmented yet. Hence we have to extend the hair and shoulder start region in order to be able to define a proper background start region, i.e. a start region that does not cover the hair or shoulder region.

Background Region For non-uniform backgrounds we have to define a background start region as shown in Figure 5b. But we do not let the background start region grow, again because the goal of the first segmentation stage has already been reached at this point. The image is segmented roughly, so the algorithm is now ready for the second segmentation stage.

(a) Hair (yellow) and shoulder (blue) start region

(b) Background start region (green)

(c) Image overlaid with regions

(d) Pure regions

Figure 5: Result after segmentation of face (5a), after first (5b) and after final stage (5c, 5d)

2.3.2 Second Segmentation Stage

In the second stage the algorithm tries to refine the rather coarse first result by letting the different regions compete against each other with relaxed parameters. Here some of the regions may disappear. This can happen in cases where a region is not actually present in the image, and hence it is displaced by the other growing regions. A common example are shoulders that are completely covered by hair.

2.3.3 Final Segmentation Stage

After upsampling the current result to the original image resolution, we again grow all regions similar to the previous step. Finally, after the growing process has converged, we have a refined version of the segmentation result on full image resolution (Figures 5c and 5d). This segmentation result is now ready for the post-processing stage. The whole work flow is summarized in Figure 6.

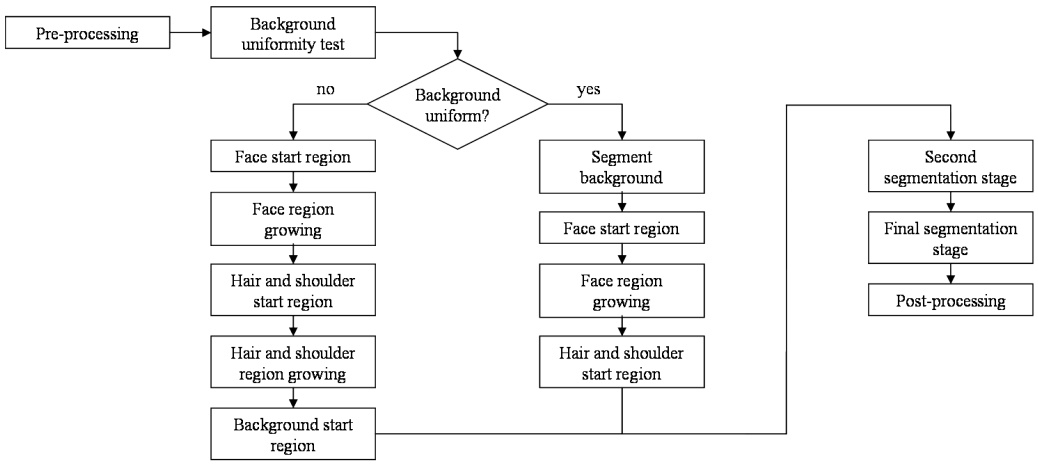


Figure 6: Work flow of our algorithm

2.4 Post-Processing

The post-processing stage enhances the segmentation result by removing or relabeling of regions. To do so, we again use prior knowledge about the scene in the image. First, improbable regions, for example face regions located near the image border, are removed. After that, we correct nested regions, i.e. regions that are completely surrounded by another region (e.g. hair surrounded by background is labeled as background).

3 Experiments & Results

In our experimental evaluation we used a dataset consisting of 320 face images. One part of these images was taken from the Caltech face database [3] and the FERET database [11], the other portion are proprietary images provided by Siemens PSE Graz, Biometrics Center. 197 of the 320 face images show a uniform background, while the background in the remaining 123 images is non-uniform. Since homogeneous backgrounds can be segmented more easily than non-homogeneous ones, the segmentation error is usually lower for images with a uniform background. All images are aligned with respect to eye locations, as this is the only constraint on the input passport photographs. Furthermore hand labeled ground truth data for all images is available. The annotated images allow us to calculate error metrics on our segmentation results, and to compare our algorithm to other methods.

3.1 Error Metrics

In order to determine the segmentation error for each region we use two error rates, the false positive and the false negative rate, which describe the difference between the segmented region and the hand labeled region relatively to the true region size:

- **False positive rate (FP):** The false positive rate describes the error of segmenting a certain region in places where this region actually does not appear in the ground truth data.
- **False negative rate (FN):** The false negative rate describes the error of not segmenting a certain region in places where this region actually does appear in the ground truth data.

To calculate the error rates we use the following formulas:

$$FP = \frac{numRegionFalsePositivePixels}{\max(numRegionPixelsGroundTruth, 0.01 \times imageArea)} \times 100 \quad [\%] \tag{10}$$

$$FN = \frac{numRegionFalseNegativePixels}{\max(numRegionPixelsGroundTruth, 0.01 \times imageArea)} \times 100 \quad [\%] \tag{11}$$

The denominator in Equations (10) and (11) has a lower bound of $0.01 \times imageArea$, which prevents the error rates from shooting up for very small regions. Furthermore it is important to bear the duality of these two error rates in mind. One can always minimize one of the rates by letting the other one grow. Hence a good performance is only given if both error rates are adequately low. For determining the segmentation error on the whole image we use the following three error metrics:

$$misclassified = \frac{numMisclassifiedPixels}{imageArea} \times 100 \quad [\%] \tag{12}$$

$$unclassified = \frac{numUnclassifiedPixels}{imageArea} \times 100 \quad [\%] \tag{13}$$

$$totalError = misclassified + unclassified \quad [\%] \tag{14}$$

3.2 Quantitative Results

In this section we compare our method to an expert system and an AdaBoost classification algorithm. We evaluated all three approaches on our dataset using the error metrics defined in the previous section. The results are outlined in the following table and Figure 7. The obtained results demonstrate that our algorithm outperforms both the expert system and the AdaBoost classifier. The expert system has the worst overall performance of all three methods. The AdaBoost classifier generally shows a higher performance than the expert system. As one can see in Figure 7, our method is quite robust and does not suffer from any extremely high error values, like the expert system and the AdaBoost classifier. This is a result of the good generalization ability of our algorithm. By using only general knowledge about typical passport photographs we avoid that our method specializes in a certain set of images. In contrast to this the expert system is rather vulnerable to failing on new image sets, because it uses very specific rules. Regarding runtime, a Matlab-implementation of our algorithm takes approximately 4 seconds (depending on the image content) to segment a 320x240 image on a standard desktop PC.

	Expert System	AdaBoost	Our Algorithm
Misclassified [%]	12.01	9.60	4.69
Unclassified [%]	1.80	0.00	0.00
Total error [%]	**13.81**	**9.60**	**4.69**

Table 1: Mean segmentation errors calculated from 320 face images

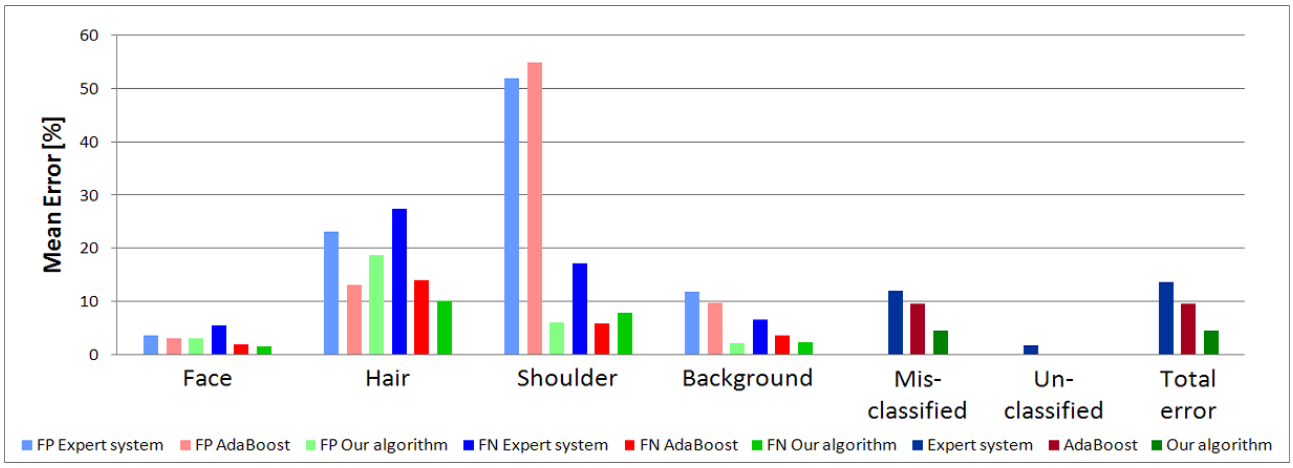


Figure 7: Comparison of our algorithm to the expert system and AdaBoost classifier. The charts present mean error rates and mean overall errors.

4 Conclusion

In this paper we presented an automatic, knowledge based segmentation algorithm that partitions pre-aligned face portrait images into face, hair, shoulder and background region. We showed that our algorithm outperforms two other methods targeted at the same problem, namely an expert system and an AdaBoost classifier. One reason for this is surely its good generalization ability. The knowledge that we incorporate into the segmentation process is general and applicable to any aligned face image. However, there is still room for future research. As one can see in Figure 7, the hair region has the highest error rates among all regions. In order to improve the segmentation of hair one might use more advanced texture descriptors, like higher order moments or Gabor filters. Also glasses can cause problems. They severely affect the definition of the hair start region, and in the final segmentation they are often labeled as hair due to color similarity. To overcome this problem a glasses detector might be integrated into the segmentation algorithm.

Acknowledgement

This work has been funded by the Biometrics Center of Siemens IT Solutions and Services, Siemens Austria.

References

[1] A. F. Abate, M. Nappi, D. Riccio, and G. Sabatino. 2d and 3d face recognition: A survey. *Pattern Recognition Letters*, 28(14):1885 – 1906, Oct 2007.

[2] X. Bresson, S. Esedoglu, P. Vandergheynst, J. P. Thiran, and S. J. Osher. Fast global minimization of the active contour/snake model. *Journal of Mathematical Imaging and Vision*, 28(2):151–167, 2007.

[3] Caltech. Caltech face database. http://www.vision.caltech.edu/html-files/archive.html, 1999.

[4] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic Active Contours. *International Journal of Computer Vision*, 22(1):61–79, February 1997.

[5] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1-2):89–97, 2004.

[6] Faculty of Electrical Engineering and Computing. Segmentation of id photographs using adaboost algorithm. Technical report, FER, University of Zagreb, Croatia, 2008.

[7] B. Fasel and J. Luettin. Automatic facial expression analysis: A survey. *Pattern Recognition*, 36(1):259–275, Jan. 2003.

[8] J. Huang and D. Mumford. Statistics of natural images and models. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 541–547, 1999.

[9] ISO International Standard ISO/IEC JTC 1/SC37 N506. Biometric data interchange formats - part 5: Face image data, 2004.

[10] M. J. Jones and J. M. Rehg. Statistical color models with application to skin detection. *International Journal of Computer Vision*, 46(1):81–96, 2002.

[11] P. J. Phillips, H. Wechsler, J. Huang, and P. Rauss. The FERET database and evaluation procedure for face recognition algorithms. *Image and Vision Computing*, 16(5):295–306, April 1998.

[12] T. Pock, M. Unger, D. Cremers, and H. Bischof. Fast and exact solution of total variation models on the gpu. In *CVPR Workshop on Visual Computer Vision on GPUs*, pages 1–8, 2008.

[13] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. CL-Engineering, 3rd edition, 2007.

[14] M. Storer, M. Urschler, H. Bischof, and J. A. Birchbauer. Face image normalization and expression/pose validation for the analysis of machine readable travel documents. In *Proceedings of OAGM/AAPR Conference*, pages 29–39, 2008.

[15] M. Subasic, S. Loncaric, and J. A. Birchbauer. Expert system segmentation of face images. *Expert Systems with Applications An International Journal*, 36(3):4497–4507, 2009.

[16] M. Unger, T. Pock, W. Trobin, D. Cremers, and H. Bischof. TVSeg - interactive total variation based image segmentation. In *Proceedings British Machine Vision Conference 2008*, Leeds, UK, September 2008.

[17] V. Vezhnevets, V. Sazonov, and A. Andreeva. A survey on pixel-based skin color detection techniques. In *Proceedings of GraphiCon*, pages 85–92, 2003.

[18] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Computing Surveys*, 35(4):399–458, Dec 2003.