Eyeglasses contour extraction using genetic algorithms

Borza Diana-Laura
Computer Science Department,
Technical University of Cluj-Napoca,
Cluj-Napoca 400114, Romania
borza_diana@yahoo.com

Adrian Darabant
Computer Science Department,
Babeş-Bolyai University,
Cluj-Napoca 400114, Romania

Radu Gabriel Dănescu
Computer Science Department,
Technical University of Cluj-Napoca,
Cluj-Napoca 400114, Romania
Radu.Danescu@cs.utcluj.ro

***Abstract*—This paper presents an eyeglasses contour extraction method that uses genetic algorithms to find the exact shape of the lenses. An efficient shape description, based on *Fourier coefficients,* is used to represent the shape of the eyeglasses, allowing a wide range of shapes to be represented with a small number of parameters. The proposed method does not require the position of the eyes to be known in advance. The conducted experiments demonstrate the effectiveness of the proposed solution.**

***Keywords*— Eyeglasses Extraction; Fourier coefficients; Genetic Algorithms.**

I. Introduction

Face analysis and recognition is one of the most active research topics in computer vision and pattern recognition due to its numerous practical applications, which include biometrics, surveillance systems, human emotion understanding etc. The majority of face detection and recognition methods need to extract several facial features (eyes, eyebrows, nose). Eyeglasses can deteriorate the performance of these systems as they occlude some features and their lenses can cause strong specular reflections. The modeling and extraction of eyeglasses from images could increase the performance of facial recognition systems.

Modern eyeglasses prescriptions require the measurement of several morphological properties of the patients face and their correlation with the morphology of the frames that the patient will wear: the interpupillary distance, the size of the frames (*boxing width and height*), the *vertex distance* (the distance between the outer surface of the cornea and the inner surface of the lens) etc. In recent years, several computer-aided systems ([1]) were developed that can accurately measure these parameters by processing digital images. To compute these parameters, the position of the pupils and the geometry of the glasses must be precisely determined in the input image.

There are several recent works that address the problem of eyeglasses detection, localization and removal.

Jiang et al. [2] defined six measures that can be used to decide whether the eyeglasses are present or not in an image. Jing et al. [3] modeled the eyeglasses using a deformable contour, and used dynamic programming to extract several points on the frames. Wu et al. [4] designed a complex trinocular system, and extracted the rims of the eyeglasses using 3D Hough Transform and additional geometrical constraints. Wu et al. [5] used a Markov Chain Monte Carlo method to determine 15 key points on the contour of the eyeglasses. A statistical analysis and synthesis approach is then used to learn the mapping between pairs of face image with or without glasses from a database. In [6] the authors proposed an eyeglasses extraction method that uses Fourier coefficients to describe the shape of the lenses; a multi-stage Monte Carlo sampling technique is used to accurately find the contour of the eyeglasses.

Genetic Algorithms (GA) have already been used to solve several pattern recognition problems. In [7] the authors proposed a method to estimate the shape parameters of hand drawn circles. Each circle is described by an (x, y, r) triplet (centroid and radius) and a GA is set up to find the best circle in the input image. Yao et al. [8] use a multiple-population genetic algorithm and evolution-clustering to detect ellipses. Ellipses are described by five elements: minor and major axis, center coordinates and the rotation angle. Ozcan et al. [9] developed a new approach for partial shape matching, by applying a genetic algorithm to an attribute string representation of polygonal shapes. Kala et al. [10] proposed an offline handwriting recognition using genetic algorithms and graph theory. The images of handwritten characters are converted into graphs, and GAs are used to match the character from the test image to the characters from a known dataset.

The rest of this work is organized as follows: Section 2 presents an overview of the proposed solution, and in Section

3 the main steps of the solution are detailed. The experimental setup and the performance evaluation results are described in Section 4. The work is concluded in Section 5.

II. SOLUTION OUTLINE

The proposed solution uses *genetic algorithms* to find the shape of the eyeglasses rims. Genetic algorithms are search heuristics that generate solutions to optimization and search problems using ideas from genetics and natural evolution: selection of the fittest, crossover and mutation. The search space is explored based on a dataset – that contains *Fourier descriptors* of the most common eyeglasses shapes – and on a linear morphing procedure between the elements of this set.

Figure 1 depicts the outline of the proposed solution. First, the population of the genetic algorithm is initialized by morphing randomly selected shapes from the eyeglasses dataset. The fitness of every individual (chromosome) in the population is evaluated by matching the shape of the eyeglasses over the distance transform of the input image. Next, individuals from the population are selected through a fitness-based process (i.e. chromosomes with a higher fitness are more likely to be selected) and their genome is modified by applying crossover and mutation operators. This process (evaluate, select, recombine, mutate) is iteratively repeated until the average relative change in the fitness function value over *s* generations is less than a user defined value, and the fittest individual form the last generation is selected as the solution to the eyeglasses extraction problem.

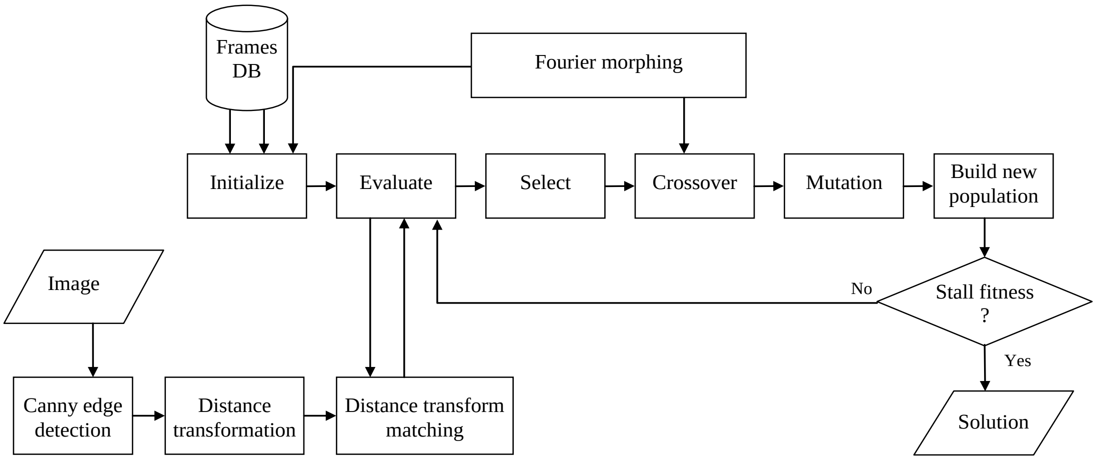

Figure 1. Solution outline

III. SOLUTION DESCRIPTION

A. Shape Description

Fourier descriptors are a way of representing 2D closed shapes by taking the *Fourier* transform of the boundary, where every contour point (x, y) is mapped to a complex number $z(i) = x(i)+ jy(j)$. The Fourier descriptors $c(k)$ are the coefficients of the Fourier transform of the shape:

$$c(k) = \sum_{i=0}^{N-1} z(i) e^{-\frac{j2\pi ki}{N}} . \tag{1}$$

The original shape $z(i)$ can be recovered by applying the inverse Fourier transform:

$$z(i) = \sum_{i=0}^{N-1} c(k) e^{-\frac{j2\pi ki}{N}} . \tag{2}$$

The $c(k)$ coefficients describe the frequency contents of the shape: low values of k describe low frequency information, while higher values describe details (Figure 2). Therefore, if only a few coefficients are used in the reconstruction, a coarse approximation of the original shape is obtained. By increasing the number of components in the description, higher frequencies are also taken into consideration, and a more detailed representation is achieved. The first two coefficients $c(0)$ and $c(1)$ describe the centroid and the size of the shape, respectively. If all the other components are set to 0, the shape becomes a circle. Higher frequency components alter the circle specified by $c(1)$.

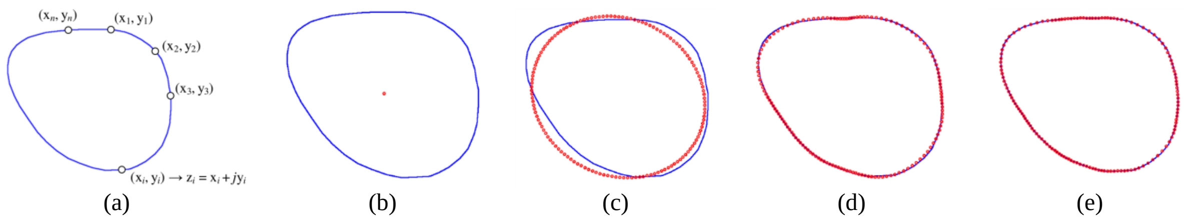

Figure 2. Shape reconstruction using Fourier descriptors (from [6]). (a) Initial shape ; (b) 1 descriptor; (c) 2 descriptors; (d) 6 descriptors; (e) 14 descriptors

As in [6] a linear morphing procedure between two existing shapes, S_1 and S_2, is used to generate a new intermediary shape:

$$S_m = \alpha S_1 + (1-\alpha) S_2 \tag{3}$$

where α is a number between 0 and 1. Figure 3 shows the result of morphing between two shapes with different values for α.

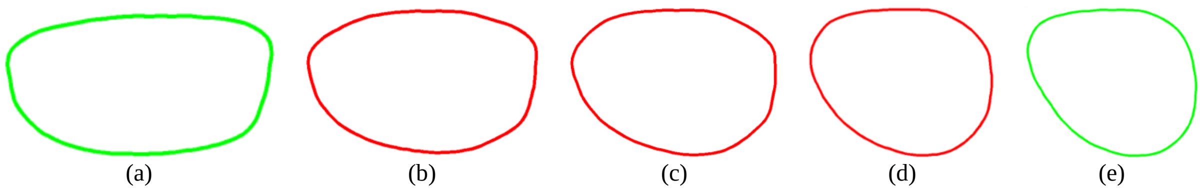

Figure 3. Morphing between two shapes using Fourier descriptors (from [6]). (a). S_1, $\alpha = 0$; (b). $\alpha = 0.25$; (c) $\alpha = 0.5$; (d) $\alpha = 0.75$; (e) S_2, $\alpha = 1$

For the eyeglasses extraction problem, the shapes of the lenses are described using the first 14 Fourier coefficients. Therefore, a relatively large number of eyeglasses shapes are described by a limited number of parameters.

B. Genetic algorithms

Genetic algorithms ([11]) are search heuristics that mimic the process of natural evolution and selection, to generate solutions for optimization or search problems. The algorithm iteratively alters a population of potential solutions (chromosomes or individuals). In each generation (iteration), individuals are randomly selected from the current population, and they are used as parents to produce new offsprings. Over the generations, the algorithm evolves to a solution to the optimization problem.

A basic genetic algorithm is described in Algorithm 1.

Algorithm 1 – Basic genetic algorithm

- Initialize the population $P(t = 0)$ with randomly generated candidate solutions (chromosomes)
- Until a termination criterion is not met (e.g. number of iterations performed, or a solution with adequate objective function value is found)
  - Evaluate population $P(t)$
  - Select parents
  - Crossover (recombine pair of parents to produce new off-springs)
  - Mutate off-springs
  - Build next population $P(t+1)$; $t \leftarrow t + 1$;
- Return fittest individual from $P(t)$

<u>Chromosome encoding</u>

One of the most important issues that need to be addressed when developing a genetic algorithm is the solution encoding. The most commonly used encoding type is binary encoding. However, due to the complexity of the search problem in question, as eyeglasses come in very different shapes, we chose a value based encoding, where each chromosome is represented by an array of values.

The following coefficients are sufficient for fully describing an eyeglasses shape:

$$c = [lx \quad ly \quad rx \quad ry \quad r \quad S]^T , \tag{4}$$

where (lx, ly), (rx, ry) are the centroids of the left and right lenses respectively, r is the radius of the lens, and S is an array containing the 14 *Fourier coefficients* that describe the shape of the left lens. The number of coefficients used to describe the eyeglasses was heuristically established. The right lens is computed by horizontally flipping the left one; this assures the symmetry of the model.

<u>Initialization</u>

The first step of the algorithm is to randomly generate the initial population based on a uniform probability distribution, assuring the coverage of the entire search space.

To randomly generate a chromosome, each gene is initialized with a uniformly distributed value $g_i \sim U(g_{low}, g_{up})$, where g_{low} and g_{up} are the lower and upper boundaries for the given gene.

The size and the centroids of the lenses are uniformly generated, taking into account the validity constraints imposed on a chromosome.

The shape of the lenses is determined based on an eyeglasses dataset. Similar to [6] the dataset was created by manually selecting the boundary of the left eyeglasses rim from 40 facial images and by computing the *Fourier descriptors* of these boundaries. The dataset was further extended with shapes that were generated by morphing between the existing shapes from the initial database. All the data was normalized to scaling and translation. To further explore the shape search space, Fourier morphing is performed between randomly extracted shapes from the database.

Fitness evaluation

In many real life situations, the contour of the eyeglasses presents multiple gaps, due to light reflections and occlusions. Therefore, for the evaluation of a chromosome, Canny edge detection [12] is performed, followed by a Distance Transformation [13] in order to propagate the response of the edge detector to nearby image locations. The distance transform is an operator that produces a grayscale image (distance map, distance field), similar to the input image, in which the gray level intensities show the distance to the closest object pixel (usually an edge point).

Figure 4 depicts the Canny edge map and the corresponding distance transform of the input image.

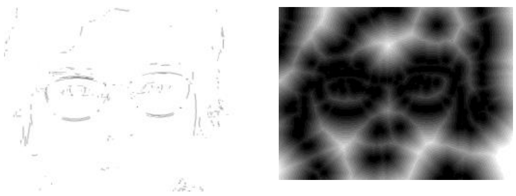

Figure 4. Canny edge detection and Distance Transform

In order to evaluate the extent to which a rim (ρ) matches the input image, its shape is superimposed over the distance transform image and the matching score is computed as the average value of all the pixels that lie under the boundary of the rim:

$$score(\rho) = \frac{1}{|\rho|} \sum_{(x,y)\in\rho} DT(x,y)\,, \tag{5}$$

where ρ is the contour of the shape and DT(x, y) is the pixel at position (x, y) from the distance transform image.

The fitness of an individual is defined as the average between the matching scores of left and the right rims:

$$fitness(chromo) = \frac{score(\rho 1) + score(\rho 2)}{2}\,, \tag{6}$$

where $\rho 1$ and $\rho 2$ are the contour of the left and right frame, respectively.

Selection

Selection is a genetic operator that chooses potentially useful parent solutions for the next generation. Roulette wheel selection uses the fitness value of an individual to compute its probability of being selected. If the fitness of a chromosome is f, its probability of being selected is $p = \frac{f}{\sum_{i=0}^{n-1} f_i}$, where n is the number of individuals in the population, and f_i is the fitness of the i^{th} individual in the population. Chromosomes with higher fitness values are more likely to be selected; therefore an individual can be selected more than once and contribute to the generation of more than one child.

It is highly probable that good individuals are lost when cross-over and mutation results in offspring that are weaker than the parents. *Elitism* is a process that preserves the best b chromosomes unchanged in the next generation. This strategy can lead to a significant improvement of the algorithm, as it guarantees that the solution quality does not decrease from one generation the next one. In the proposed solution, the first b = 2 best individuals are left unchanged from one generation to another one.

Crossover

Crossover and mutation are the two operators that can be used to modify the genetic material (Figure 5).

Crossover (recombination) simulates biological reproduction: the process involves taking two or more parent solutions as input, and producing new solutions (offsprings) by performing some operations between the parents. The frequency with which the crossover operator is applied is controlled by the *crossover rate* parameter. This parameter is directly proportional to the number of new individuals that are added to the population. However, if this rate is too high, “good” solutions can be discarded faster than selection can produce improvements. On the other hand, a low value of this parameter can cause the population to stagnate, as there is little exploration of the search space.

Several types of crossover were implemented: *one point crossover, uniform crossover* and *arithmetic crossover.* For one point crossover, a single crossover point on the parents is selected, and to produce the offspring, the genes from the beginning to the crossover point are copied from one parent, and the rest is copied from the second parent. In uniform crossover, the genes of the offspring are randomly selected from the two parents. This means that the recombination is done at gene level and not at segment level, as in one point crossover.

The best results were obtained using arithmetic crossover. Arithmetic crossover involves applying an arithmetic operation on the selected genes to produce a new offspring. To implement this type of recombination, a random number *r* between (0, 1) is generated and the offspring is the result of the linear interpolation between the parents with factor *r*. In the case of the gene that encodes the shape of the eyeglasses, the resulting gene is computed by morphing the shapes of the parents with factor *r*.

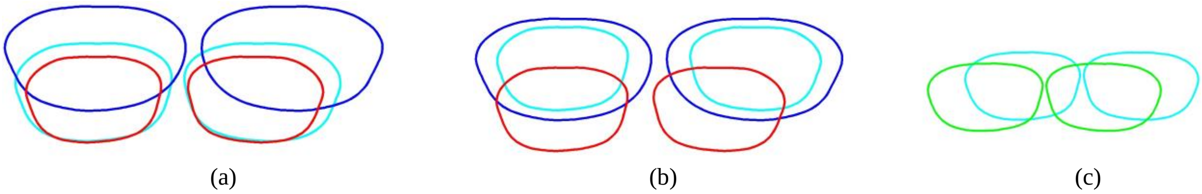

Figure 5. Genetic operators. The parent individuals are depicted in blue and red, and the offspring in cyan (a) Arithmetic crossover (linear interpolation between the genes of the parents); (b) Uniform crossover (the offspring has the frames position from the "blue" parent and the shape and the size of the lenses from the "red" parent); (c) Gaussian mutation (in the image, the position of the frame is altered by a Gaussian probability distribution centered in the previous position)

Mutation

Mutation is defined by a unary operator and it is used to inhibit premature convergence and to maintain the genetic diversity. The mutation operator is applied to a genotype, by altering the value of one or more genes from the chromosome, and produces a mutant, modified version of it. Similar to crossover, mutation occurs according to a user defined *mutation rate*. This probability should be set to a low value (usually 0.01); if it is set too high the genetic algorithm becomes a primitive random search.

Three types of mutation operators were tested: *boundary*, *uniform* and G*aussian mutation.* The boundary mutation operator replaces the gene with either the lower or the upper bound of the gene. Uniform mutation sets the value of the selected gene with a uniform random number selected between the upper and lower bounds, and Gaussian mutation replaces the value of the gene with a Gaussian distributed random value centered in the initial value of the gene.

Figure 6 presents the evolution of the genetic algorithm population and the selected solution.

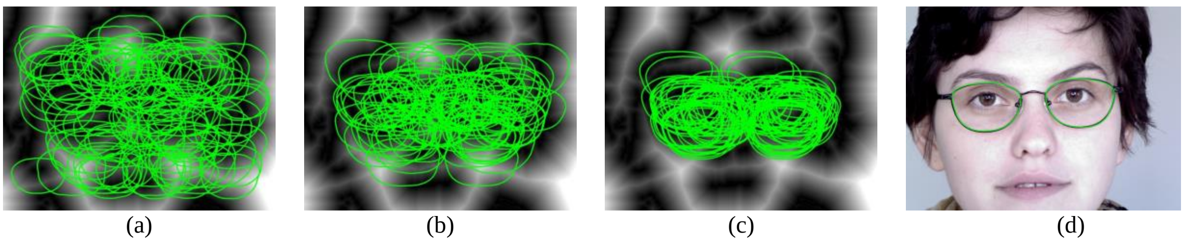

Figure 6. Iterations of the genetic algorithm and selected solution

IV. Experimental Results

The proposed solution is targeted for computed aided optometrical systems, which measure morphological parameters needed for modern eyeglasses prescriptions. The experimental setup is the following: the patient stands in front of the device, at a distance ranging from 1 to 2 meters and a high resolution facial image is captured. The test environment is controlled: therefore the algorithm assumes that the eyeglasses are present in the input image, and that the input images do not contain extreme face pose variations.

The solution was tested on 63 images which were captured in real life conditions by optometrical measurement systems [1]. Each image has an xml file attached which stores the position of the eyes, the points of the contour of the eyeglasses (manually marked by the optician), and the measured value of several optometrical parameters. For the testing procedure, we used the same metric as in [6]: the distance between the extracted contour and the contour marked by the optician (the ground truth). The output of the algorithm is considered a true positive if the overlapping ratio between the extracted rim and the ground truth is larger or equal to 0.95. A recognition rate of 93.65% was achieved. Some eyeglasses rims extraction results are shown in Figure 7. The algorithm does not yield good results on rimless eyeglasses, as the matching procedure is based only on edge information. Some failure cases are depicted in Figure 8.

In the next part of this section, a comparative study between the proposed solution and the other methods reported in the literature is performed. However, several works focus on eyeglasses detection [2], or on eyeglasses' removal [5]. Therefore, a relevant comparison with all the other methods is not possible. In [3] the authors propose an eyeglasses contour extraction algorithm, but the testing dataset is not publically available. Other methods [4] use a complex trinocular system to detect the eyeglasses.

A comparable method is proposed in [6]: the eyeglasses are represented using Fourier descriptors; the eyes are detected and a multistage Monte Carlo sampling procedure is to find the eyeglasses contours in a region around the eyes.

Our approach uses the same method to describe the shape of the eyeglasses. In addition, the eyeglasses' database is further extended using the proposed morphing procedure in order to obtain a higher diversity in the first steps of the algorithm.

As opposed to [6] our approach does not require the position of the eyes to be known in advance. The lens of the eyeglasses can present strong specular reflection, and the eyes are often impossible to detect under these conditions. In conclusion, our approach can localize the eyeglasses even if the eyes are not detected. The search heuristic is much simplified than the one proposed in [6]. In [6] first a Monte Carlo sampling procedure is used to generate eyeglasses hypotheses; next these potential solutions are clustered in order to determine the regions that are more likely to contain the eyeglasses, and finally a random search Monte Carlo step is used to search the best solution the previously detected regions. In our solution, the genetic algorithm population naturally converges to the optimal solution over the iterations of the algorithm.

Note: A video illustrating the evolution of the population on several test images can be downloaded from: https://drive.google.com/folderview?id=0Bwn_NrD78q_TfkN5WTFpYVAtbG9ROXpNWFptX3J5elI1dVpYN3NRbUVCZ0xpdVpXUk5STlk&usp=sharing .

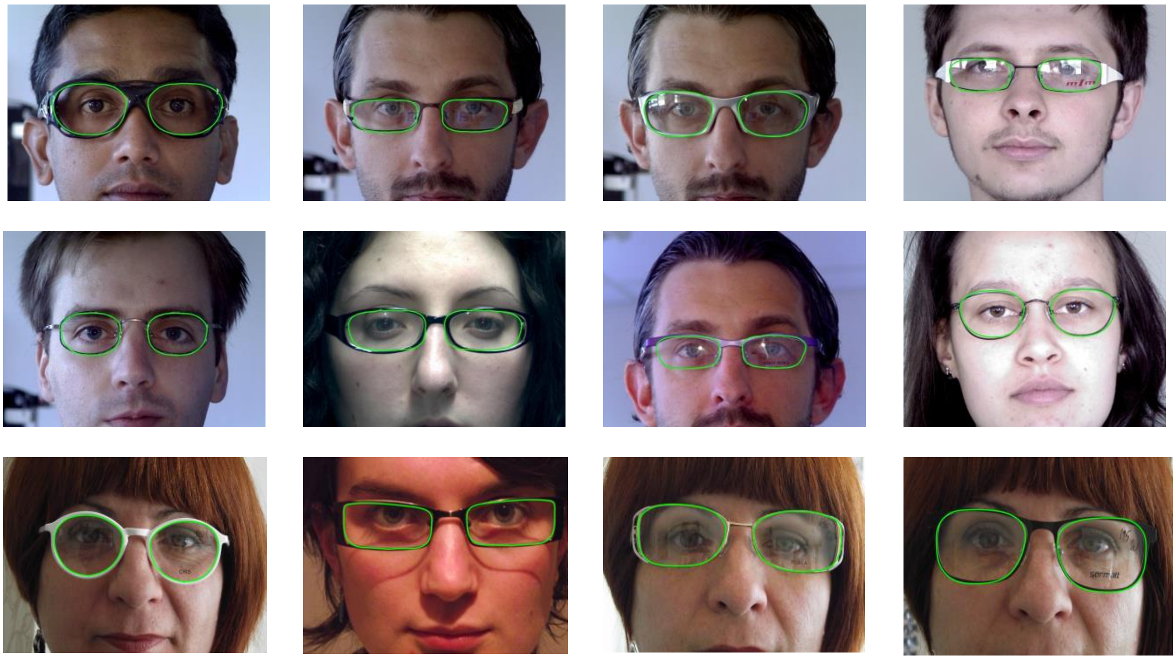

Figure 7. Eyeglasses contour extraction results

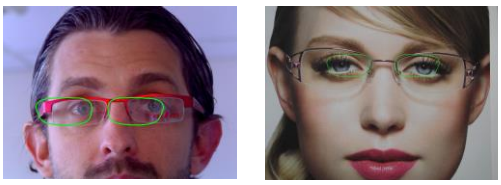

Figure 8. Failure cases

V. Conclusion

This paper presented an automatic eyeglasses contour extraction system from facial images. The algorithm uses an efficient shape description based on *Fourier coefficients*, which allows the representation of various eyeglasses shape using only 14 variables. The eyeglasses extraction problem is resolved using a *genetic algorithm*, were each chromosome encodes the shape of the eyeglasses. The initial population is generated based on the Fourier coefficients form a dataset that contains the most representative eyeglasses shapes. The search space is further explored using a linear morphing procedure that allows the generation of new intermediate shapes from two input shapes. The performance rate of 93.65% demonstrates the effectiveness of the proposed solution.

As future work, we shall develop a more sophisticated method for matching a candidate solution over the input image, as the current method is based solely on edge information. We also plan to push our system to eyeglasses contour extraction in video sequences.

REFERENCES

[1] ACEP France homepage. 2015. Available from http://www.smart-mirror.com/fr [21 May 2015]

[2] Jiang, X., Binkert, M., Achermann, B., & Bunke, H., 2000. Towards detection of glasses in facial images. *Pattern Analysis & Applications*, *3*(1), pp. 9-18.

[3] Jing, Z., & Mariani, R., 2000. Glasses detection and extraction by deformable contour. *Proceedings of the 15th International Conference on Pattern Recognition*, Barcelona, Portugal, pp. 933-936.

[4] Wu, H., Yoshikawa, G., Shioyama, T., Lao, S., & Kawade, M., 2002. Glasses frame detection with 3D Hough transform. *Proceedings of the 16th International Conference on Pattern Recognition,* Quebec, Canada, pp. 346-349.

[5] Wu, C., Liu, C., Shum, H. Y., Xy, Y. Q., & Zhang, Z., 2004. Automatic eyeglasses removal from face images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *26*(3), pp. 322-336.

[6] Borza, D., Dărăbant, A. & Dănescu, R., 2013. Eyeglasses Lens Contour Extraction from Facial Images Using an Efficient Shape Description. *Sensors,* 13(10), pp. 13638-13658.

[7] Ayala-Ramirez, V., Gasca-Martinez, J. A., Sanchez-Yanez, R. E., & Mota-Gutierrez, S. A., 2008. *Parametric Circle Approximation Using Genetic Algorithms*. INTECH Open Access Publisher.

[8] Yao, J., Kharma, N., & Grogono, P. 2005. A multi-population genetic algorithm for robust and fast ellipse detection. *Pattern analysis and applications*, *8*(1-2), pp. 149-162.

[9] Ozcan, E., & Mohan, C. K., 1997. Partial shape matching using genetic algorithms. *Pattern Recognition Letters*, *18*(10), pp 987-992.

[10] Kala, R., Vazirani, H., Shukla, A., & Tiwari, R., 2010. Offline handwriting recognition using genetic algorithm. *arXiv preprint arXiv:1004.3257*.

[11] Holland, J., 1972. *Adaptation in Natural and Artifcial Systems.* Ann Arbor, University of Michigan Press.

[12] Canny, J., 1986. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 8(6), pp. 679-698.

[13] Fabbri, R., Costa, L. D. F., Julio C. Torelli & Odemir M. Bruno, 2008. 2D Euclidean Distance Transform Algorithms: A Comparative Survey. *ACM Computing Surveys,* 40(1), pp. 1-44.