Improvement of a Face Detection System by Evolutionary Multi-Objective Optimization

Rodrigo Verschae, Javier Ruiz del Solar
Universidad de Chile, Dept. Electrical Engineering
Av. Tupper 2007, 837-0451 Santiago, Chile
rodrigo@verschae.org, jruizd@cec.uchile.cl

Mario Köppen, Raul Vicente Garcia
Fraunhofer IPK, Dept. Security Technologies
Pascalstr. 8-9, 10587 Berlin, Germany
{mario.koeppen|raul}@ipk.fraunhofer.de

Abstract

This paper presents the application of evolutionary multi-objective optimization (EMO) to the improvement of a face detection system. The face detection system is based on the boosted cascade system, and analyzes image positions on different scales in a three-step-procedure. Based on threshold settings, the algorithm decides whether to continue with the test on a finer scale at the current position. Thus, the thresholds for all scales and stages have a major influence on the performance of the system, and become the subject of the evolutionary optimization according to three objectives: low false positive rate, high detection rate and low processing time. The used EMO is the extension of the Standard Genetic Algorithm to the EMO case by using Fuzzy Pareto Dominance as a meta-heuristic. The application of this EMO to the face detection system is explored and discussed using images from a standard face detection benchmark dataset. From the runtime analysis of the EMO it can be concluded that the algorithm reliably approaches the Pareto set of the problem.

1. Introduction

Face detection is a key step in almost any computational task related with the analysis of faces in digital images (access control, identification for law enforcement, borders control, identification and verification for credit cards and ATM, human-machine interfaces, passive recognition of criminals in public places or buildings, etc.). Given an arbitrary image, the goal of a face detection system is to find all contained faces and to determine the exact position and size of the regions containing these faces. When analyzing real-world scenes, face detection is a challenging task, which should be performed robustly and efficiently (real-time), regardless variability in scale, location, orientation (in-plane rotation), pose (out-of-plane rotation), illumination, artifacts (beard, eye-glasses, etc.), and facial expressions, and considering possible object occlusions.

In this paper, we are considering the optimization potential of a face detection system, which has a cascading architecture. The heuristic used to optimize the system corresponds to find the thresholds of a grid designed for avoiding classifying every possible window of the image in this cascade, thus reducing processing time by preserving low error rates. This problem is of multi-objective nature. While it is common to consider a weighted summarization for the two error rates in detection systems (false positive and false negative rate), the influence of such an internal feature reduction is not so directly linked to these error rates, and should rather be considered an objective by its own. It is well-known that in a multi-objective optimization problem the set of *supported solutions* (i.e. optima obtained from a convex combination of the objectives as a weighted sum) is usally not covering the Pareto front once this front contains concave parts. In case of combinatorial multi-objective optimization, the ratio of the number of supported solutions to the number of elements of the Pareto set may even fall exponentially with problem dimension (see [16] for this happening with the multi-objective 0/1 knapsack problem).

Noting that a complex image analysis system usually contains a large set of adaptable parameters, where the influence in the objectives (once they are e.g. computed by using a set of test data) can hardly be modeled directly, it has to be considered that only an exploration of the Pareto front of the system can yield sufficient insight into the suitability of such a system for its application. Here, we study the use of an Evolutionary Multi-Objective Optimization algorithm in a given face detection system. In this system, three objectives are minimized: (1) false positive rate, (2) false negative rate and (3) number of evaluated features, which influences the speed of the detector. It is worth to remember that by minimizing the false negative rate (FNR), the detection rate (DR) is maximized (DR=1-FNR).

Section 2 recalls the basic concepts of EMO and introduces the used algorithm, which is based on a fuzzification of the Pareto dominance relation. The reason to chose this algorithm here were: (1) scale-invariance in the objectives, (2) implicit handling of the crowding problem, and (3) ease of implementation, since it is a simple extension of the Standard Genetic Algorithm. Then, section 3 presents the face detection system, and the manner in which it was made subject to multi-objective optimization. Section 4 studies the results of the application of the EMO onto this system, before some conclusions are drawn in section 5.

2. Basic Concepts

2.1. Fuzzy-Pareto-Dominance

In multi-objective (or multi-criterion) optimization, the optimization goal is given by more than one objective to be extreme. Formally, given a domain as subset of R^n, there are assigned m functions $f_1(x_1, \ldots, x_n), \ldots, f_m(x_1, \ldots, x_n)$. Usually, there is not a single optimum but rather the so-called Pareto set of *non-dominated* solutions:

For two vectors a and b it is said that *a (Pareto) dominates b*, in terms $a >_D b$, if each component of a is less or equal to the corresponding component of b, and at least one component is smaller:

$$a \ >_D \ b \longleftrightarrow \forall i(a_i \leq b_i) \wedge \exists k(a_k < b_k). \tag{1}$$

Note that in a similar manner Pareto dominance can be related to $>$-relation.

The subset of all vectors of a set M of vectors, which are not dominated by any other vector of M is the Pareto set (also Pareto front). The Pareto set for univariate data (single objective) contains just the maximum of the data.

We consider the fuzzification of the Pareto dominance relation, as given with the following definition, in order to base a multi-objective extensions of single-objective evolutionary algorithms on it [7]:

It is said that vector *a dominates* vector b by degree μ_a with

$$\mu_a(a,b) = \frac{\prod_i \min(a_i, b_i)}{\prod_i a_i} \tag{2}$$

and that vector *a is dominated by* vector b at degree μ_p with

$$\mu_p(a,b) = \frac{\prod_i \min(a_i, b_i)}{\prod_i b_i} \tag{3}$$

Remarks: Note that the definitions differ in the denominator and thus are not symmetric: "dominating by degree μ" and "being dominated by degree μ" have different fuzzy values. The definition is similar to so-called subsethood degrees as introduced by Kosko [8].

For a Pareto-dominating b, $\mu_a(a,b) = 1$ and $\mu_p(b,a) = 1$, but $\mu_p(a,b) < 1$ and $\mu_a(b,a) < 1$.

We may use these dominance degrees to rank a set M of multivariate data (vectors) like the fitness values of a multiobjective optimization problem. Each element of M is assigned the maximum degree of being dominated by any other element of M, and the elements of M are sorted according to the ranking values in increasing order:

$$r_M(a) = \max_{b \in M \setminus \{a\}} \mu_p(a,b) \tag{4}$$

Note that this definiton is related to a set. A *ranking value* of a within M can only be assigned with reference to a set M containing a.

By sorting the elements of M according to the ranking values in increasing order (FPD ranking, FPD for Fuzzy-Pareto-Dominance), we obtain a partial ranking of the elements of M. For vectors with the same ranking values (like all dominated vectors), we have to assign a random ordering. There is no additional cue for complete ranking of these vectors.

An important property of this ranking scheme is that it can not be obtained by sorting of a weighted sum of the components. More general, it can be shown that there is no scalar function of vector components of one vector at all, which will give the same ranking of the vectors of a set M. This can be shown by a simple counterexample. In addition it should be noted that the FPD ordering is also scale-invariant.

2.2. SGA$_{f2r}$ Algorithm

Fuzzy-Pareto-Dominance can be considered a meta-heuristic to make single-objective optimization algorithms capable of handling multiple objectives as well.

This is achieved by using the ranking values of the fitness objectives vectors as replacements of the fitness values in the single-objective case. The main prerequisite is the suitable identification of sets M, for which the ranking values are specified. In evolutionary computation, M will be easily identified with the population of individuals.

If we apply this concept to the *Standard Genetic Algorithm* (SGA) we obtain the SGA_{f2r} algorithm, where the subscript $f2r$ indicates the replacement of fitness values by ranking values. Due to space limitations, for more details of the algorithm the reader is referred to [7].

3. Face Detection System

3.1. System Overview

Several approaches have been proposed for the computational detection of faces in digital images. Starting with the seminal works or Rowley [12] and Sung&Poggio [14], successful proposed approaches include the use of neural networks [2], SNoW classifiers [19], Bayesian classifiers [13], SVM [11], and boosted cascades [15][10]. A comprehensive review can be found in [5][18]. Nowadays, most successful results have been obtained by using boosted cascades and neural approaches [2][15][17][3][10][6][9][4].

The here implemented face detection system is based on the boosted cascade system proposed by Viola and Jones [15], with the later improvements proposed by Wu et al. [17]. These two cascade face detectors outperform previous systems in terms of processing speed, by keeping a high recognition rate. Our face detection system employs simple rectangular features (a kind of Haar wavelets) [15] and LBP-based features [3], a nested-cascade of filters [17] that discard non-face images, the integral image for a fast computation of the rectangular features [15], weak partitioning real Adaboost as a boosting strategy for the training of the detectors [17], and LUTs (Look-up Tables) for a fast evaluation of the weak classifiers [17]. It is worth to mention that using real Adaboost, a confidence value of each detected face is calculated. That means that the detection results are not binary (yes/no) but real-valued. The key idea to obtain a fast detection is that the complexity of the filters (i.e number of features) increases when advancing in the cascade. In this way for windows that are easy to classify, not much processing is performed.

The block diagram of the face detector is presented in figure 1. First, a multiresolution analysis is performed by scaling the input image by a factor of 1.2 (Multiresolution Analysis module), for detecting faces at different scales. This scaling is performed until images of about 24x24 pixels are obtained. Afterward, in the Window Extraction module, for each of these scaled versions of the input image, windows of 24x24 pixels are extracted. The extracted windows are then further processed by a cascade of real Adaboost filters (10 stages). Each of these filters feed back information, the result of the filter processing (confidence value of been a face) to the Window Extraction module. This information is employed for deciding if the image, in the corresponding window position, needs to be further processed in a lower scale (resolution) or not.

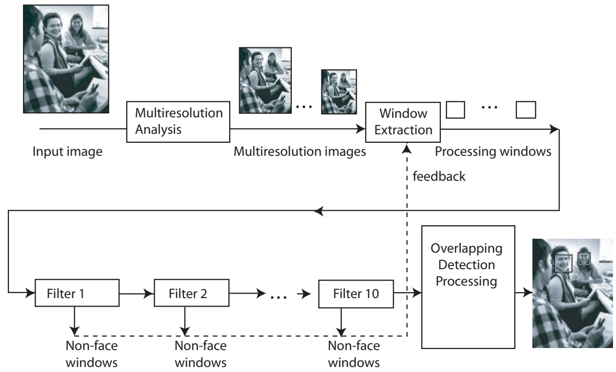

Figure 1. Block diagram of the face detection system.

When a window is classified as non-face by any of the filters non-further processing is done. In the opposite case, the window goes further in the cascade. A window is classified as face if and only if all filters of the cascade classify it as a face, i.e. it passes through all cascade filters. After all selected windows are processed and classified as faces or non-faces, in the Overlapping Detection Processing module the face windows are analyzed and fused (normally a face can be detected at different scales) for determining the size and position of the final detections. In this module, the confidence values associated to the detections are used for deleting or fusing detections.

In this article the parameters of the Window Extraction and Overlapping Detection Processing modules are optimized. It is important to notice that this optimization is done after finishing the training of the filter cascade.

3.2. Multiresolution and Parameters' Optimization

As already mentioned, images are analyzed in different resolutions by scaling them by a factor of 1.2, until images of about 24×24 pixels are obtained. Each level of the pyramid is analyzed in three stages, in a coarse to fine manner (this heuristic was proposed in [4]). In a first stage (s1) only image positions on a sparse grid with grid step step$_1$=6 are used for extracting windows. In this way only about 2.8% of all possible positions are evaluated. Each grid position with a score value (the confidence value feed back by the filters) below a threshold th_{s1} is a starting point for a local refinement of the search. In the second stage (s2) a finer structured grid around each starting point of the coarse grid is evaluated, using a grid step step$_2$=3. Then, each grid position with a score value below a threshold th_{s2} is evaluated in the third stage (s3) using a 3×3 neighborhood. The speed up of the process is controlled by the threshold's values th_{s1} and th_{s2}. Higher threshold's values means higher processing speed (less fine scales are evaluated), but lower detection rates (many true faces can be lost).

When analyzing and fusing face windows in the Overlapping Detection Processing module two other threshold's values needs to be employed. On the one hand, if the number of overlapped face windows in a given position is larger than th_{num}, then they are considered as a true detection and fused. On the other hand, if the detection volume of the overlapped face windows in a given position is larger than th_{vol}, then they are considered as a true detection and fused. The detection volume is defined as the sum of all confidences values corresponding to the overlapped windows.

Considering that a cascade with 10 filters is implemented, altogether 22 parameters are to be adjusted by the EMO. Each filter has associated two parameters, th_{s1} and th_{s2}. The remaining two parameters are th_{sum} and th_{vol}.

4. Results

The SGA$_{f2r}$ algorithm was applied to the parameter setting of the face detection system. In the system, 22 parameters were identified that control the selection of the masks in the cascade. After selection of the parameters, three objectives were computed:

1. **False Positive Rate**: Relative number of non-faces that the system has wrongly output a positive face detection.
2. **False Negative Rate**: Relative number of face the system did not detect as face.
3. **Number of Features**: Number of features that were taken into account for the face detection decision. This is related to the number of filter being applied.

	Run 0	Run 1	Run 2	Run 3	Run 4
N	40	20	20	20	40
μ_m	0.1	0.05	0.05	0.05	0.05
μ_c	0.4	0.9	0.9	0.9	0.9
p	0.2	0.3	0.1	0.9	0.9

Table 1. Five setting for SGA$_{f2r}$ algorithm: N: Population size; μ_m: mutation rate; μ_c: cross-over rate; p: elitism ratio.

Of course, these objectives are in conflict: the fewer objective 3, the more the chance is increasing to base the face detection decision on an insufficient data evaluation, thus increasing FPR and FNR. Between FPR and FNR the existence of a trade-off is a well-known fact.

Each of the 22 parameters was encoded as an integer using 8 bit. Then, SGA$_{f2r}$ was trained with a suite of 20 images from the MIT and CMU dataset [12] (containing 191 faces) and various parameter settings over 200 generations. Table 1 lists some of the choices that were explored. `Run 0` is a highly explorative setup: mutation rate is comparable high, population size is larger, and elitism, i.e. the relative number of parent individuals taken unmodified into the next generation (sorted by their ranking values in the parent population), is small. Compared to `run 0`, `run 1` is using the cross-over operator more often to generate new individuals, and this ratio is even more increased in `run 2`. `Run 3` and `run 4` are more conservative, with an elitism of 0.9. Here, `run 4` is using the larger population (40 individuals).

All runs yield comparable Pareto sets (see fig. 2 for an example). These Pareto fronts demonstrate the conflict in the objectives, and reveal their seemingly convex shape. `Run 4` seems to provide the best coverage of the Pareto front. An easy analysis showed that various parameter settings for the points of the Pareto fronts already improve the formerly-used manual set-up of the face detection system. Majorly, a slight reduction in the number of features was achieved, while the FNR could be reduced without increasing the FPR.

For further analysing the result (and providing an attempt to the tricky question about the true achievement here) we have analyzed the run-time behaviour of the algorithm runs. Fig. 3 gives the changes of the *crowding measure* over the generations. The crowding measure was computed in a manner similar to the one used in the NSGA-II algorithm [1]: Given is a set P of n data points P_i of m dimensions P_{ij} each. Each P_i for each dimension j has a closest neighboring data point P_i^l and P_i^r to its left and right according to the j-th coordinate values of all data points (these data points may differ for different dimensions). The points on the "border", i.e. having the smallest or largest value in a dimension, are ignored in the computation. The crowding measure is computed as

$$c(P) = \sum_i \prod_j (P_{ij}^r - P_{ij}^l) \quad (5)$$

The crowding measure gives an empirical measure for the progress of the Pareto set of the algorithm (as stored in the archive) towards the Pareto front of the objective space. Generally, a larger value indicates lower crowding and better diversification of the algorithm on the Pareto front. If the measure drops during the evolution, this could be seen as an indicator for "gap filling" between the non-dominated individuals. If it remains nearly constant, this is an indicator of progress of the whole front. A jump in the measure could be seen as the addition of a non-dominated outlier to the archive.

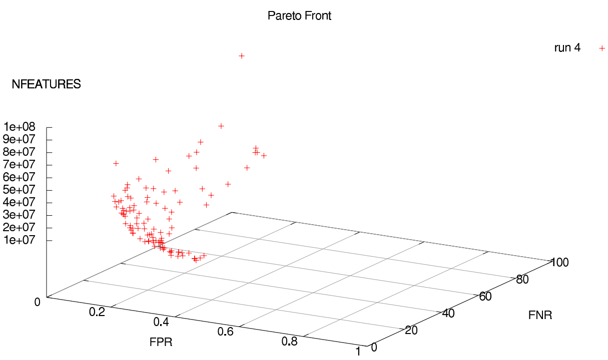

Figure 2. Pareto front of a run of SGA$_{f2r}$ algorithm on the face detector.

Comparing fig. 3 with the corresponding increase in size of the archive, as shown in fig. 4, the best performance of the setup for `run 0`, i.e. the highly explorative variant, can be seen clearly: while collecting more non-dominated individuals, the crowding measure is kept nearly constant. We also see on the opposite that the conservative version `run 4` is increasing the archive even more, but the crowding measure is becoming smaller. This version more and more fills the gaps between the already found non-dominated points. A smaller population (`run 3`) helps to increase the crowding measure. Worst are the variants employing cross-over more often than mutation (`run 1` and `run 2`) showing lower sampling at the Pareto front, and also a lower crowding measure.

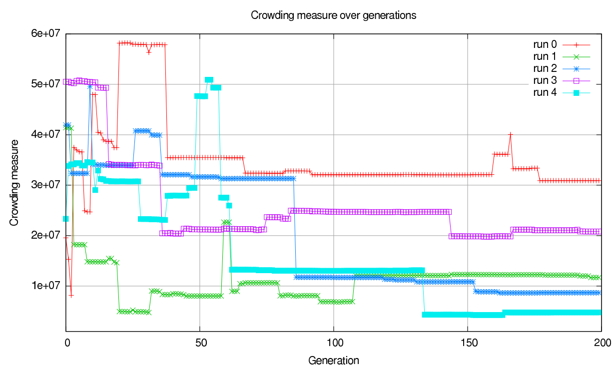

Figure 3. Evolution of crowding measure (same as used in NSGA-II) shows the intrinsic crowding control of SGA$_{f2r}$ algorithm.

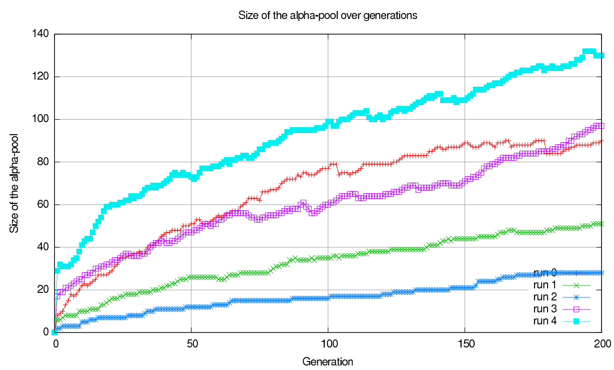

Figure 4. Evolution of archive sizes during the 200 generations.

Jumps in the crowding measure are more common up to generation 100, together with a larger growth in the archive size. After generation 100, the archives start to grow more slowly, while the crowding measures tend to be more constant. Thus, the initial explorative stage can be distinguished from a slow convergence against the true Pareto front (at least the one accessible by the algorithm) later on. Given this, and the convex shape of the found Pareto sets, the task of improving the face detector by means of multi-objective optimization shows no pecularities. We may conclude on a stable convergence against the Pareto front of the face detection system.

5. Conclusions

In this paper we studied the application of evolutionary multi-objective optimization (EMO) to the improvement of a face detection system. The considered face detection system is based on the boosted cascade, and it analyzes image positions on different scales in a three-step-procedure. Based on threshold settings, the algorithm decides whether to continue with the test on a finer scale at the current position. Thus, the thresholds for all scales and stages have a major influence on the performance of the system. Accordingly, 22 thresholds parameters were identified and became the subject of the evolutionary optimization according to three objectives: low false positive rate, high detection rate and low processing time. The EMO that was used in the optimization is an extension of the Standard Genetic Algorithm to the EMO case by using Fuzzy Pareto Dominance as a meta-heuristic. The application of this EMO to the face detection system was explored and discussed using test images from a standard face detection benchmark dataset. From the runtime analysis of the EMO, esp. by using a crowding measure, and the archive size, it can be concluded that the algorithm reliably approaches the Pareto set. The conclusion is based on the following observations: the Pareto fronts found at different settings of the algorithm parameter appear to be convex and similar in shape; and the EMO showed clearly an explorative stage at the beginning, followed by a converging global approach to the Pareto front. The optimized system achieves especially improvements in the obtained processing speed (by reducing the number of considered features) and false positives rates.

References

[1] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J. J. Merelo, and H.-P. Schwefel, editors, *Proceedings of the Parallel Problem Solving from Nature VI Conference*, volume 1917 of *Lecture Notes in Computer Science*, pages 849–858, Paris, France, 2000. Springer.

[2] M. Delakis and C. Garcia. Convolutional face finder: A neural architecture for fast and robust face detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(11):1408–1423, 2004.

[3] B. Fröba and A. Ernst. Face detection with the modified census transform. In *Int. Conf. on Face and Gesture Recognition FG 2004*, pages 91–96, 2004.

[4] B. Fröba and C. Küblbeck. Robust face detection at video frame rate based on edge orientation features. In *Int. Conf. on Automatic Face and Gesture Recognition FG 2002*, pages 342–347, 2002.

[5] E. Hjelmøas and B. Low. Face detection: A survey. *Computer Vision and Image Understanding*, 83:236–274, 2001.

[6] L. Huang, A. Shimizu, and H. Kobatake. Classification-based face detection using gabor filter features. In *Int. Conf. on Face and Gesture Recognition FG 2004*, 2004.

[7] M. Köppen, R. Vicente Garcia, and B. Nickolay. Fuzzy-pareto-dominance and its application in evolutionary multi-objective optimization. In *EMO 2005, LNCS 3410*, pages 399–412. Springer, 2005.

[8] B. Kosko. *Fuzzy Engineering.* Prentice-Hall, Upper Saddle River, NJ, 1997.

[9] Y. Lin, T. Liu, and C. Fuh. Fast object detection with occlusion. In *Proc. ECCV 2004*, 2004.

[10] C. Liu and H. Shum. Kullback-leibler boosting. In *Proceedings CVPR 2003*, 2003.

[11] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: an application to face detection. In *Proceedings of CVPR'97*, 1997.

[12] H. Rowley, S. Baluja, and T. Kanade. Neural network-based detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(1):23–28, 1998.

[13] H. Schneidermann and T. Kanade. A statistical model for 3d object detection applied to faces and cars. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 746–751, 2000.

[14] K. Sung and T. Poggio. Example-based learning for viewed-based human face deteccion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(1):39–51, 1998.

[15] P. Viola and M. Jones. Fast and robust classification using asymmetric adaboost and a detector cascade. In *Advances in Neural Information Processing System 14*. MIT Press, 2002.

[16] M. Visée, J. Teghem, M. Pirlot, and E. Ulungu. Two-phases method and branch and bound to solve the bi-objective knapsack problem. *Journal of Global Optimization*, 12(2):139–155, 1998.

[17] B. Wu, H. AI, C. Huang, and S. Lao. Fast rotation invariant multi-view face detection based on real adaboost. In *Int. Conf. on Face and Gesture Recognition FG 2004*, pages 79–84, 2004.

[18] M. Yang, D. Kriegman, and N. Ahuja. Detecting faces in images: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(1):34–58, 2002.

[19] M. Yang, D. Roth, and N. Ahuja. A SNoW-based face detector. In *Advances in Neural Information Processing Systems 12*, pages 862–868. MIT Press, 1999.