Feature Fusion Hierarchies for Gender Classification

Fabien Scalzo[1], George Bebis[2], Mircea Nicolescu[2], Leandro Loss[2]
[1] Geffen School of Medicine, University of California, Los Angeles
[2] Computer Vision Lab, Department of Computer Science, University of Nevada, Reno
fscalzo@mednet.ucla.edu

Abstract

We present a hierarchical feature fusion model for image classification that is constructed by an evolutionary learning algorithm. The model has the ability to combine local patches whose location, width and height are automatically determined during learning. The representational framework takes the form of a two-level hierarchy which combines feature fusion and decision fusion into a unified model. The structure of the hierarchy itself is constructed automatically during learning to produce optimal local feature combinations. A comparative evaluation of different classifiers is provided on a challenging gender classification image database. It demonstrates the effectiveness of these Feature Fusion Hierarchies (FFH).

1. Introduction

The generalization of new image acquisition devices and the development of new feature appearance extractors have recently increased the interest of combining complementary modalities to perform complex image classification tasks. The fusion of different feature sets is promising for a large extent of applications in medical imaging, biomedical, remote sensing, robotics and computer vision. Hierarchical approaches [5, 3, 2] to image classification are particularly interesting to solve complex problems because they are capable to decompose them into tasks that are often easier to tackle. However, those approaches often tend to manually define the structure of their hierarchy depending on the features involved [7], and can only exploit a limited number of features.

The current paper addresses these problems by presenting a framework, named Feature Fusion Hierarchies (FFH) (Section 2), that performs image classification based on a large set of features extracted from Gabor and Laplace filters. The learning of optimal Feature Fusion Hierarchies (FFH) is a particularly challenging task because both the *structure* and the *parameters* of the hierarchy have to be estimated. To this end, we propose to exploit a genetic learning algorithm (Section 3) to explore the space of possible hierarchies.

The effectiveness of the framework is evaluated in Section 4 on a gender classification task from facial images, which is a two-class image classification problem. Results are compared to the state-of-the-art by reporting the accuracy of different classifiers used within the framework: Nearest Neighbors Classifiers (NN), Linear Discriminant Analysis (LDA), Support Vector Machines (SVM) and Kernel Spectral Regression (KSR).

2. Feature Fusion Hierarchies

Feature Fusion Hierarchies (FFH) address the problem of fusing high-dimensional registered feature sets for image classification. The representational framework takes the form of a two-level hierarchy which combines local feature fusion and decision fusion into a unified model (Figure 1).

Given a feature set $I(x, y, f)$, where (x, y) denotes a position in the image, and f is a feature, the feature fusion level is defined as a set of compound features $\mathcal{C}$. Each compound feature $\mathcal{C}_i$ combines a subset of features $f_{\mathcal{C}_i}$ over a local window $\theta_{\mathcal{C}_i}$. This fusion is performed using a dimensionality reduction technique, and denoted $\mathcal{R}_i(I_{f_{\mathcal{C}_i},\theta_{\mathcal{C}_i}})$. It is learned a supervised way (*e.g.* LDA). A key property of this function $\mathcal{R}_i$ is to operate locally in the sense that it exploits local adaptive windows [4] whose parameters $\theta_{\mathcal{C}_i} = \{x, y, Sx, Sy\}$ are automatically adjusted during learning (position in the image (x, y), width Sx and height Sy). The output of the function $S_i = \mathcal{R}_i(I_{f_{\mathcal{C}_i},\theta_{\mathcal{C}_i}})$ corresponds to a vector of lower dimensionality. An additional classifier is learned on the top of the first level to form the second level $\mathcal{D}$ corresponding to the decision fusion. Its input data correspond to the compound feature output $\{S_1, S_2, \ldots, S_n\}$ merged into a single vector S.

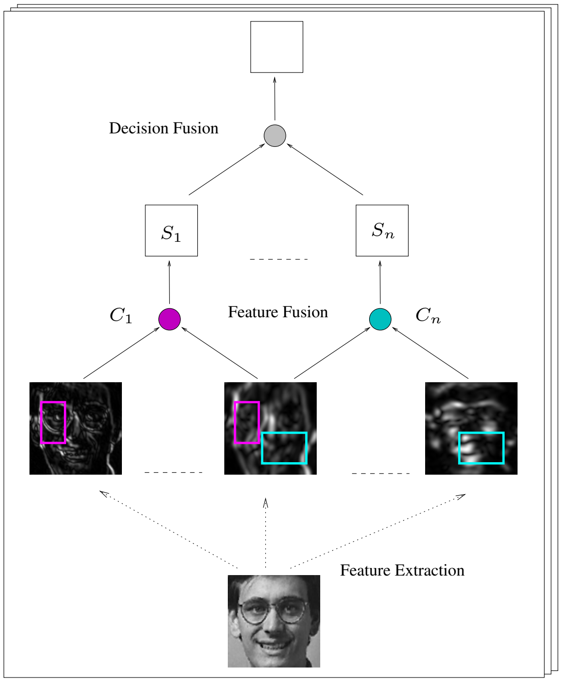

Figure 1. Overview of a Feature Fusion Hierarchy (FFH). For a given image, Gabor and Laplace features are extracted and used as input to local feature fusion operators C_i. The second level classifier exploits the responses S_i to produce a single decision value.

3. Learning of Fusion Hierarchies

Learning Feature Fusion Hierarchies is particularly challenging because both the *structure* and the *parameters* of the model have to be estimated. Unlike many existing methods, we neither manually assign which features to combine nor the local regions of interest on which the system should focus. Instead, we rather let the system learn what is the optimal hierarchy in terms of discriminative power.

The proposed method uses an evolutionary approach (summarized in Algorithm 1) to explore the space of possible hierarchies. The optimal solution is the one that offers the best classification accuracy on the validation data while minimizing the number of features used and the total area covered by the patches. In the following, we discuss the encoding of genomes (Section 3.1), the fitness function (Section 3.2) as well as the crossover and mutation strategies (Section 3.3) of the proposed GA, respectively.

Algorithm 1 Genetic Learning Algorithm

// Generate the initial population
$\{\mathcal{P}\} \leftarrow \text{generate}()$
// Evaluate the fitness of each genome $g \in \mathcal{P}$
$\mathcal{F} \leftarrow \text{eval}(\mathcal{P})$
for each i $<$ nIteration **do**
  // Rank-based selection
  $\mathcal{P}' \leftarrow \text{select}(\mathcal{P}, \mathcal{F})$
  // Crossover
  $\mathcal{P}' \leftarrow \text{combine}(\mathcal{P}')$
  // Mutate
  $\mathcal{P}' \leftarrow \text{mutate}(\mathcal{P}')$
  // Evaluate
  $\mathcal{F}_c \leftarrow \text{eval}(\mathcal{P}')$
  // Create the new population using elitism
  $\mathcal{P} \leftarrow \text{generate}(\mathcal{F}, \mathcal{F}_c, \mathcal{P}', \mathcal{C})$
end for

3.1. Genome Representation

Each evolving individual (*i.e.* genome) in the population is represented as a binary vector encoding the structure and the parameters of a specific hierarchy.

A genome defines the hierarchy as a set of N_c combinations $\mathcal{C}_i$. Each combination in the genome is defined by two parts:

- The *structure* corresponds, for each combination in the hierarchy, to the subset of features that are combined $f_{\mathcal{C}_i} = \{f_1, \ldots, f_n\}$.
- The *parameters* $\theta_{\mathcal{C}_i} = \{x, y, Sx, Sy\}$ of a combination define the spatial position x, y and size Sx, Sy of the local window in the image on which the fusion is performed.

As illustrated in Figure 2, a genome encodes these two types of information, structure and parameters, into a single binary vector. Given n features at the first level, the structural part is represented as a n-length vector encoding the presence of the features in the combination. Each bin is associated with one bit whose value is 1 is the corresponding feature is a part of the combination. For the parameter part, variables $\{x, y, Sx, Sy\}$ are each represented as b bits vector. Parameter b is chosen such that x, y cover the entire image. During learning, an additional constraint insures that x, y are in the image coordinate and Sx, Sy are strictly positive.

3.2. Fitness

The goal of the optimization framework is to find the structure and the parameters of the hierarchy that performs best in terms of classification accuracy. Therefore, the fitness function $\mathrm{fit}(g)$ is set proportional to the classification rate r of the genome encoded hierarchy g,

$$\mathrm{fit}(g) = r(g) + \alpha_1 n + \alpha_2 s^{-1} \tag{1}$$

where n is the number of zeros in the structure part of the genome g and s is the total area covered by the patches. Parameters α_1 and α_2 are used respectively to support combinations that both have a fewer number of features and are defined over a smaller window.

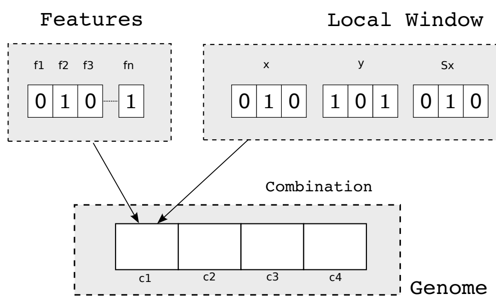

Figure 2. A Feature Fusion Hierarchy made of four compound features ($c1, c2, c3, c4$) is encoded into a genome. The structural part and the parameters are embedded into a single binary vector.

3.3. Mutation and Crossover

The crossover operator uses individuals in the population that have been selected to generate offsprings. A bi-parental crossover is used in our algorithm to produce new individuals.

Given N individuals selected in a linear Rank-based strategy, the algorithm produces a new population of the same size. To this purpose, two parents are picked at random from the pool of individuals and used in the crossover operation. This is done using the result of a random crossover operator. Each bit of the vector is randomly chosen (with probability $1/2$) from one of the two parent vectors. The second offspring consists of the components not chosen for the first resulting vector.

The mutation operation is performed using a single point mutation operator which picks a mutating gene randomly with the mutation probability using a biased coin toss. After the mutating gene is selected, its value is inverted.

3.4. Elitism

An elitist strategy is used to prevent from losing the best solutions found at each iteration. The best 10 chromosomes are copied to the population in the next generation.

4. Experimental Evaluation

The effectiveness of the proposed framework is now evaluated on a gender classification task. Given a set of facial images captured under various conditions, the task is to correctly identify the gender (Male or Female) of the subject present in the image.

Gender classification is particularly interesting because it is one of the most important visual tasks for human beings. Social interactions critically depend on the correct gender perception of the parties involved. The automatic learning of the visual features that are relevant for this task is therefore particularly interesting and challenging.

The dataset and the protocol used in our experiments are described in Section 4.1. In Section 4.2, we present the Gabor and Laplace convolutions that are used to extract different features on each image and thus produce the input data of the Feature Fusion Fierarchies. Quantitative results for different classification techniques are compared to the state-of-the-art and discussed in Section 4.3.

4.1. Dataset

The dataset used in our experiments is the same that has been used by SUN *et al.* in their comparative study [6]. It contains 400 distinct frontal images and is particularly challenging due to the presence of different races, facial expressions, and lighting conditions. The 400 images were equally divided between males and females. Some of these images are illustrated in Figure 3. As experimental protocol, the average error rate was recorded using a three-fold cross-validation procedure with the restriction that the number of male and female faces must be equal in each separate training, validation and test set.

As it was mentioned in [6], it must be noticed that this database is more challenging than those used in other studies, where the same person appears multiple times in the dataset or where the intraclass variation is low.

4.2. Feature Extraction

Each image is convolved by a set of Gabor and Laplace filters to produce features that constitute the input data of our framework. A particularity of Gabor Filters is to share some characteristics with certain cells of

Figure 3. Illustration of the Gender Classification Database [6] used during our experiments. The images are particularly challenging due to the large intra-class variation.

the visual cortex. In addition these filters have demonstrated improved accuracy in many computerized visual tasks [7]. A Gabor Filter is a linear filter whose impulse response is defined by a harmonic function multiplied by a Gaussian function,

$$g(x,y) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \psi\right) \quad (2)$$

$$x' = x\cos\theta + y\sin\theta \qquad y' = -x\sin\theta + y\cos\theta \quad (3)$$

where λ is the wavelength of the cosine factor, θ represents the orientation of the normal to the parallel stripes of a Gabor function in degrees, ψ is the phase offset in degrees, and γ is the spatial aspect ratio that specifies the ellipticity of the support of the Gabor function.

A number of 35 Gabor Filters and 5 Laplacian Filters were used to convolve each image and creating the first level feature set.

4.3. Results

For the purpose of these experiments, we compare the effectiveness of four different classifier, namely nearest neighbors NN, support vector machine (SVM), linear discriminant analysis (LDA) and kernel spectral regression KSR [1].

Both KSR and SVM techniques exploit kernel projection, also known as the "kernel trick", to use the linear classifier to solve a nonlinear problem by mapping the observations into a higher-dimensional space, where the linear classifier is subsequently used. In our experiments, a Radial Basis Function (RBF) kernel is used as a projection matrix,

$$K(x_i, x_j) = \exp^{(-\gamma ||x_i - x_j||^2)}, \gamma > 0 \quad (4)$$

The classification results after a three-fold cross-validation are reported in Table 1 for different classifiers. It can be observed that the use of our Feature Fusion Hierarchies (FFH) reduces significantly the classification error of a PCA-based framework and outperforms the results obtained by PCA-GA approach [6]. This can be explained by the fact that the Feature Fusion Hierarchies exploit local features whereas PCA-GA computes the projection on the full image. The best results are obtained using a recently developed spectral regression technique [1], SVM classifiers comes second. The number of compound features used during these experiments was automatically selected by repeating the classification for a different number of parts until no significant improvement was observed. This occurs typically around 15 compound features.

	NN	LDA	SVM	KSR
PCA[6]	17.7%	14.2%	8.9%	-%
GA-PCA[6]	11.3%	9%	4.7%	-%
FFH	**10.9%**	**7.2%**	**4.3%**	**3.8%**

Table 1. Results for three different classifiers are reported for PCA, GA-PCA [6] and the Feature Fusion Hierarchies (FFH).

5. Conclusion

We presented Feature Fusion Hierarchies (FFH) that combines a two-level fusion framework with powerful local adaptive patches. Both the structure and the parameters of the models are learned using an evolutionary learning algorithm. Our experimental results exceed the best published results, and highlight the contribution of our generic framework.

References

[1] D. Cai, X. He, and J. Han. Efficient Kernel Discriminant Analysis via Spectral Regression. In *Proc. 2007 Int. Conf. on Data Mining (ICDM'07))*, 2007.

[2] Y. Kim and I. Oh. Classifier ensemble selection using hybrid genetic algorithms. *Pattern Recogn. Lett.*, 29(6):796–802, 2008.

[3] I. T. Podolak. Hierarchical classifier with overlapping class groups. *Expert Syst. Appl.*, 34(1):673–682, 2008.

[4] F. Scalzo and J. Piater. Adaptive patch features for object class recognition with learned hierarchical models. In *Beyond Patches*, pages 1–8, 2007.

[5] R. Singh, M. Vatsa, and A. Noore. Hierarchical fusion of multi-spectral face images for improved recognition performance. *Information Fusion*, 9(2):200–210, 2008.

[6] Z. Sun, G. Bebis, X. Yuan, and S. J. Louis. Genetic Feature Subset Selection for Gender Classication: A Comparison Study. In *IEEE International Conference on Image Processing*, 2002.

[7] X. Tan and B. Triggs. Fusing gabor and lbp feature sets for kernel-based face recognition. In *Analysis and Modeling of Faces and Gestures*, pages 235–249, 2007.