Patch-based Image Classification through Conditional Random Field Model

Giuseppe Passino
Queen Mary, University of London
Mile End Rd
London, E1 4NS, UK
giuseppe.passino@elec.qmul.ac.uk

Ebroul Izquierdo
Queen Mary, University of London
Mile End Rd
London, E1 4NS, UK
ebroul.izquierdo@elec.qmul.ac.uk

ABSTRACT

We present an image classification system based on a Conditional Random Field (CRF) model trained on simple features obtained from a small number of semantically representative image patches. The CRFs are very powerful to handle complex parts dependencies due to their approach based on the effective modelling of the source probability conditioned on the evidence data, and they have been applied successfully to image classification and segmentation tasks in presence of a large number of low level features. In this paper an agile system based on the application of CRFs to images coarsely segmented is introduced. The main advantage of the system is a reduction in the required training time, slightly sacrificing the classification accuracy. The model implementation is described, experimental results are presented and conclusions are drawn.

1. INTRODUCTION

In the last years the request for ubiquitous access to multimedia content has been constantly increasing. This fact has raised issues related to how such data can be browsed and retrieved according to their *semantics*, that is, their actual content. The Content-Based Image Retrieval (CBIR) is the application of the computer vision theory to the image retrieval problem. The goal of the CBIR systems is to analyse, classify and retrieve images based on their content. However, there is no direct correspondence between image features and the semantics of the image itself. Additionally, the chosen features have to be descriptive, easy to retrieve and representative. This is particularly true if the extraction and the analysis of the features is executed in performance-critical scenarios, such as on mobile devices or on servers demanded to handle requests from a number of clients.

Local features related to specific image parts (or *patches*) can be used for the classification. On one side this approach endows a strong discriminative power due to the locality of the employed information. On the other side, however, the number of low-level local features makes a fast analysis of the data nearly impossible. In this circumstance a solution can be given by coarser-grain semantically richer features.

The part-based models have been shown to be highly performing: one of the outstanding works in this field, presented by Sivic and al. [14], is based on the *bag-of-features* system, which is the equivalent for the multimedia content of the *bag-of-words* used for text classification. Considerable results have been reached even without exploiting any information related to the relative position of the parts in the images, which is a strong source of knowledge particularly when considering semantically meaningful features.

In this paper an alternative approach is presented. It is based on a promising model that, despite being introduced quite recently, has exposed optimistic results in the image classification area: the Conditional Random Field (CRF).

CRFs have been introduced by Lafferty at al. in 2001 [5], in relation to sequences classification problems. The use of CRFs in image analysis scenarios allows the representation of the features related to single patches and to pairs of patches, so that information like the mutual distance and location among the features can be exploited. CRFs can handle quite complex system dependencies as a result of the particular probabilistic model employed. This leads to a very flexible framework that is however limited to some extent by the huge computing power required for the learning phase, especially when complex features are involved, due to the combinatorial explosion of training parameters.

The system proposed in this paper applies the CRF framework to simple colour-based features extracted from image patches obtained via a coarse-grain segmentation. The first result is an agile classification system whose training time is small compared to the current approaches. Another purpose of this paper is to investigate the effectiveness of CRFs in describing the relationships between a small number of structurally complex image components.

2. CRF IN IMAGE CLASSIFICATION

In the following Section 2.1 the data labelling problem is briefly introduced, and the CRF approach to the solution is presented. The currently available applications to the image classification field are discussed in Section 2.2.

2.1 CRF Fundamentals

Consider the problem of having a collection of objects composed of n different parts, each of them belonging to a category. The goal is to assign the correct label to each part according to an observation, that is, a measurement on the object. So, for each object to be labelled, let $\mathbf{y} = \{y_i\}, i \in$

Mobimedia'07, Month 8, 2007, Nafpaktos, Aitolokarnania, Greece.

$[1, n]$, be a n-dimensional vector associated to a configuration of the n unknown labels $y_i \in \mathcal{Y}$, and let $\mathbf{x}$ be a vector of observations. Traditional generative models such the Hidden Markov Models (HMM) would model this problem by the estimation of the joint probability $p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x}|\mathbf{y})p(\mathbf{y})$, therefore requiring the modelling of the source probability $p(\mathbf{y})$ and the relationship between the observed data and the unknown labels configuration, $p(\mathbf{x}|\mathbf{y})$. This approach requires the generation of all the $|\mathcal{Y}|^n$ possible state configurations $\mathbf{y}$ (being $|\mathcal{Y}|$ the cardinality of $\mathcal{Y}$), which in the real situations are often an intractable number. To avoid this, strict independence restrictions on the features related to different patches have to be introduced in order to perform the inference in an optimised iterative way.

On the other hand, a CRF directly models the conditional probability $p(\mathbf{y}|\mathbf{x})$ [15, 5]. This hypothesis significantly reduces the problem complexity: the source statistical distribution has not to be modelled anymore, and possibly incorrect assumptions made in this task are avoided. Additionally, a problem structured in this way does not require the independence constraints introduced in the generative models to prevent the explicit generation of all the possible state configurations. This means that the CRF is actually a more powerful probabilistic model, in terms of dependencies that can be handled, than the equivalent generative one.

Formally, a CRF is an undirected graphical model that associates to each node a random variable y_i conditioned on the evidence $\mathbf{x}$. Therefore we define a CRF as an undirected graph $G = (\mathcal{V}, \mathcal{E})$, in which every vertex $V_i \in \mathcal{V}$ is associated to a discrete random variable y_i which satisfies the Markov property in respect to G. This implies that every link between two graph nodes represents a direct dependence between the correspondent random variables. The more the graph is connected, the more direct dependencies between the variables are representable and the higher is the complexity of the model, which depends on the dimension of the graph cliques. Furthermore, the presence of cycles in the graph complicates the probabilities estimation phase.

2.2 Applications

Recently there have been successful attempts to apply the CRF theory to the image classification field [4, 12, 13]. One of the first works is by Kumar [4] and is primarily focused on the CRF problem definition for image classification applications. The problem of the features choice is not tackled in depth. The input image is simply segmented in a regular grid of rectangular patches to be labelled, without giving to the patches a definite semantic meaning.

The system presented in [12] addresses the problem of masking the patches labelling process. The *hidden variables* CRF is introduced, in which the random variables associated to the graph nodes are not the output of the process, but they constitute a hidden layer between the evidence $\mathbf{x}$ represented by the features extracted from the images and the category c of the images. This approach is loose in the definition of the labels because the patch categories are not explicitly assigned, but just the number of labels is fixed. The matching of the label classes with particular traits of the patches is performed during the learning phase, that becomes more difficult and unstable because less constrained (the labels not being specified in the training set) but for the same reason potentially more powerful. The paper does not explore the problem of the features selection; instead, the local, fine-grained SIFT descriptors [7] are used.

Finally, in [13] a model for the concurrent segmentation and labelling of the images via a pixel-based CRF is presented. The work addresses the problem of the features choice, that are designed to explicitally fit the CRF model. However, the approach described is related to the image segmentation as the system tries to perform the labelling of every pixel in the image. The resulting CRF is very complex, since it associates a random variable and a node in the graph to every pixel in the image and the training of such a system is a computationally intensive task.

3. FEATURES EXTRACTION

The system presented in this work exploits the representative power of the CRF to perform a fast image classification. The complexity of the approach is reduced by using a small number of patches and simple feature descriptors. This leads to a relatively small graph to train, where the parts labelling probabilities can be calculated in a short time. In this section the features choice is addressed as well as the extraction of a low number of semantically representative image parts.

As comparative data, figures from SIFT descriptors used in [12] can be considered. In a typical image from the "faces" category of the Caltech 101 dataset [3], the SIFT extractor produces around 2500 - 3000 different keypoints, each of which representing a random variable in the probabilistic model. The features associated to each keypoint for a SIFT descriptor form a vector $f \in \mathbb{R}^{132}$. The features are composed of geometrical- and visual-related data. The latter is essentially associated to the image gradient values in an area close to the centre of the keypoint, represented in a rotation invariant and luminance invariant (normalised) form. In particular, colour information is not involved.

Colour information, if correctly extracted and properly exploited, can offer enough information to obtain a reasonably good and fast classification in a number of situations.

3.1 Image Segmentation

The images segmentation, in order to obtain the patches to be labelled, has been performed using a colour-based anisotropic diffusion, a technique aimed at segmenting an image while preserving its semantic information.

The *anisotropic* (or *nonlinear*) *diffusion* technique has been introduced by Perona and Malik [10] as a proposal to address the issue of the "semantically meaningful segmentation of images" through luminance analysis. This task, in order to make sense, has to rely on an unequivocal definition for "semantic meaning" of the segmentation process. Actually, a semantically meaningful segmentation can be defined as a process that isolates different objects represented into images. The objects are not always discriminable via a low-level feature such the image luminance, because different values of this feature are not necessarily linked to different objects. Nevertheless in many practical scenarios the luminance (or, for multidimensional valued images as in our case, the colour) can be quite representative of the semantic objects.

The anisotropic diffusion process is a scale-space algorithm: it is applied iteratively and a more coarsely segmented image is produced at each iteration. The desirable properties for this algorithm are the causality, which means that no new regions are introduced while going towards coarser scales; the immediate localisation, related to

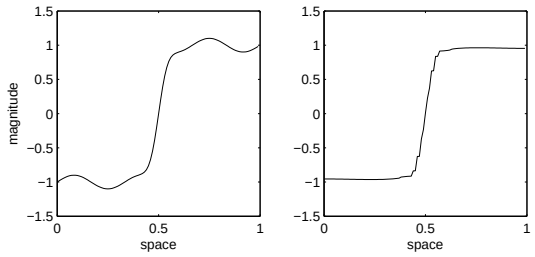

Figure 1: Effect of the application of a nonlinear diffusion filter to a monodimensional signal: on the left image the original signal shape is sketched, and on the right image the result of the filtering process when the convergence is reached.

the sharpness of the region boundaries; and the piecewise-smoothing, which implies that the intraregion smoothing should be preferred to the interregion smoothing, for every individual iteration.

In order to satisfy the previously stated constraints the algorithm implements a nonlinear smoothing by the means of an anisotropic diffusion process. The anisotropic diffusion equation is given by

$$\frac{\partial I(\mathbf{x},t)}{\partial t} = \nabla(c(\mathbf{x},t)\nabla I(\mathbf{x},t)) \ , \tag{1}$$

where $I(\mathbf{x},t)$ represents the image at different scales ($I(\mathbf{x},0)$ being the original image), $\mathbf{x}$ is a point on the image, t is the image scale, and c is the non-constant diffusion coefficient. The differential equation is discretised and iteratively solved on t (as explained in Section 4.1 and with more detail in [10]) until convergence to an image segmented in homogeneous intensity areas. Eq. (1) differs from the heat equation, which is equivalent to the application of a Gaussian blurring filter [1], because the diffusion coefficient (corresponding to the spreading of the gaussian blurring filter) is not constant in the image. The image should be strongly smoothed (big c magnitude) where the luminance does not change considerably, while it should not be smoothed (small values of c) where the luminance presents strong changes. This is achieved by choosing

$$c(\mathbf{x},t) \triangleq g(\|\nabla I(\mathbf{x},t)\|) \ , \tag{2}$$

where g in Eq. (2) is a monotonically decreasing function that has to be chosen depending on the image's structure. In this paper, it is

$$g(\|\nabla I(\mathbf{x},t)\|) \triangleq \frac{1}{1+\left(\frac{\|\nabla I(\mathbf{x},t)\|}{k}\right)^2} \ . \tag{3}$$

The choice of k in Eq. (3) is a major issue because this parameter has a large influence on the quality of the result and it is dependent on the single processed image. However, some estimations can be done to adapt the parameter depending on the specific image, as stated later in this section. The effect of the anisotropic diffusion filter is exemplified in Figure 1.

The Perona-Malik algorithm was originally developed for monochrome images, and the extension to colour images is not straightforward. The colour components are not independent, and the application of the equation to each separate component produces poor results because the semantic information lies in all the colour channels considered as a whole.

Lucchese and Mitra in 2001 proposed an algorithm to apply the anisotropic diffusion to colour images [8]. In their work a separate application of the nonlinear diffusion algorithm to the achromatic and chromatic components is proposed, as suggested by biological vision systems. The separate processing of the achromatic and chromatic components of the image has his rationale in the fact that they usually carry two different types of information.

The colour space used for the colour anisotropic filtering is the 1976 CIE Lu^*v^* [11], because it is perceptively uniform to the human vision system, and it defines a way to separate the luminance information (L) from the chromaticity information (u^*, v^*).

The luminance diffusion is performed by a standard one-dimensional nonlinear diffusion algorithm, while the chromatic components are considered as real and imaginary part of numbers in the complex space. In this way Eq. (1) can be solved in the complex domain. Since the diffusion constant c is real, Eq. (1) can be splitted in

$$\begin{cases} \frac{\partial \Re\{I_c(\mathbf{x},t)\}}{\partial t} = \nabla(c(\mathbf{x},t)\nabla\Re\{I_c(\mathbf{x},t)\}) \\ \frac{\partial \Im\{I_c(\mathbf{x},t)\}}{\partial t} = \nabla(c(\mathbf{x},t)\nabla\Im\{I_c(\mathbf{x},t)\}) \end{cases} , \tag{4}$$

where I_c is the chromatic image. Even if it is not explicit in the previous formulae, the Eq. (4) are not independent, because they are correlated through the diffusion coefficient c.

3.2 Patches Description

The segmentation via the colour-based anisotropic diffusion algorithm produces a set of image patches with almost homogeneous colour. A simple and reasonably good feature set that has been extracted from these patches for the purposes of this work is composed of the patch's colour (represented as a triplet of numbers in the RGB space) and the number of patch's pixels. The number of the pixels of a patch is introduced to let the model weight differently the patches in relation to their size. This can narrow the effect of the noise in the segmentation process, which can be partially associated to small patches. Therefore the proposed selection produces four-dimensional feature vectors.

4. SYSTEM IMPLEMENTATION

The system is divided into an image processing and segmentation module aimed at the extraction of the feature vectors and a learning module that implements the CRF model, in a two-blocks cascade system.

4.1 Feature Vectors Extraction

The naïve implementation of the nonlinear diffusion, described in [10], is used. This implementation is computationally expensive although there are different optimised versions of the algorithm available to reduce this hurdle, as for example in [16]. The algorithm works iterating over the "time" variable t and performing a first-order discretisation of the gradient function in Eq. (1) (see [10] for details).

The major conceptual difference with the original implementation of the filter is the choice of a dynamic k for Eq. (3), as suggested in [8]. This modification is a simple way to cope with the problem of the choice of the parameter k

Figure 2: On the left, an image from the "faces" category of the Caltech 101 database; on the right, the same image after the processing described in Section 4.1.

pointed out in the Section 3.1. However this modification goes further, because the parameter k is adapted in each step of the algorithm, both for the luminance and the chrominance equations, using an optimised value for each step. At each iteration a k is chosen that is equal to a given percentage p of the maximum value of the image gradient magnitude, $k = p \cdot \max_{i,j}(\|\nabla I(\mathbf{x}, t)\|)$, where I represents the achromatic or chromatic image for the two different equations solved. This choice is motivated by the fact that k in Eq. (3) plays the role of scale factor for the gradient magnitude, and comparing with the maximum magnitude is a solution to tune the filter response to the variation scale of the particular image. In our work the empirically obtained value $p = 0.01$ yields satisfactory results.

Another relevant parameter in the segmentation process is the number of iterations to be carried out. Tests have shown that a stable result can be achieved in about 3000 iterations. When the complete convergence is not acquired, the regions are not homogeneous in colour, but some smoothing is present inside them. This smoothing can be removed by the application of a colour quantisation filter.

If the complete convergence has not yet been achieved by the nonlinear quantisation process, however, the colour quantisation can introduce errors originating additional segments in the images. This effect has been reduced by the application of a discretisation filter that segments the regions removing the smoothing by considering two (four-connected) pixels as belonging to the same region if their Euclidean distance is below a certain small threshold. This approach is motivated by the fact that the nonlinear quantisation, even if not fully converged, produces sharp edges between different regions, that avoid the propagation of a region between its boundaries. The algorithm is idempotent, and the fully converged image is a fixed point for it. That is, being $\mathcal{J}$ the algorithm,

$$\begin{array}{l} \mathcal{J}(\mathcal{J}(I(\mathbf{x}, t))) = \mathcal{J}(I(\mathbf{x}, t)) \\ \mathcal{J}(I(\mathbf{x}, +\infty)) = I(\mathbf{x}, +\infty) \end{array} \quad . \tag{5}$$

In Figure 2 an example of the segmentation process output is given.

4.2 CRF Implementation

The Conditional Random Field used in this work is based on [12], so that a layer of automatically labelled hidden variables is used. There is no need for the manual annotation of every pixel in the training and test set. The graph used for the description is a tree, because in this way the solution of the CRF is easier since it becomes possible to use an exact belief propagation algorithm [2] to calculate the marginal probabilities for the nodes of the graph. The tree is obtained by running a minimum spanning tree algorithm among the patches, having assigned to each connection between two nodes a weight equal to the spatial distance between the centres of the regions. This solution is motivated by the consideration that two close patches are in average more strongly correlated than two distant ones. The construction of the CRF starting from the segmented image is shown in Figure 3.

The graph structure and the number of nodes is different for each image, but this is not an issue for the learning stage, because the parameters that determine the CRF behaviour are not dependent on the graph structure.

4.2.1 Problem Definition

Formally, the problem can be stated in this way: for each image I_i, $i \in [1, N]$, in the training set there are m_i nodes $V_{i,j}$, $j \in [1 \dots m_i]$, each of which has an l-dimensional feature vector $\mathbf{x}_{i,j}$ associated, and a class $h_{i,j}$ chosen from a set $\mathcal{H}$ of n parts. The set of feature vectors and part labels in a graph are indicated as $\mathbf{X}_i = \{\mathbf{x}_{i,1} \dots \mathbf{x}_{i,m_i}\}$ and $\mathbf{h}_i = \{h_{i,1} \dots h_{i,m_i}\}$ respectively. Additionally, a known image class $y_i \in \mathcal{Y}$ is associated to each training image. The conditional probability to have an image class y and a part labelling configuration $\mathbf{h}$ in the CRF framework is modeled as

$$p(y, \mathbf{h}|\mathbf{X}, \theta) = \frac{e^{\Psi(\mathbf{X}, y, \mathbf{h}, \theta)}}{Z(\mathbf{X}, \theta)} \quad , \tag{6}$$

where θ is the parameters vector of the CRF to be optimised during the training and

$$Z(\mathbf{X}, \theta) = \sum_{\mathbf{h}, \mathbf{y}} e^{\Psi(\mathbf{X}, y, \mathbf{h}, \theta)} \tag{7}$$

is the normalisation factor. The function Ψ, also called *local function* or *compatibility function* [15] is the core of the CRF descriptive power, being the function that embeds all the dependencies among the part labels and between the labels and the features. The general form of Ψ is

$$\Psi(\mathbf{X}, y, \mathbf{h}, \theta) = \sum_{c \in \mathcal{C}(G)} \psi(c, \mathbf{X}, y, \mathbf{h}_{|c}, \theta) \quad , \tag{8}$$

where $\mathcal{C}(G)$ is the set of cliques on G, ψ is a real-valued function, and the syntax $\mathbf{h}_{|c}$ means the portion of $\mathbf{h}$ associated to c, or the projection of $\mathbf{h}$ on c.

When the graph is a tree, the cliques are represented by the nodes themselves and the pairs of nodes connected by an edge. The functions associated to a single node are commonly referred to as *state functions* in the literature, while the ones referring to linked pairs of nodes are named *edge functions*. Moreover, usually the dependence on the CRF parameters θ is linear in order to simplify the form of the likelihood gradient during the optimisation; the edge functions do not depend on the a-priori knowledge $\mathbf{X}$; and the state functions are affected only by the value of the feature vector related to the node to which they refer. The form

$$\begin{array}{rl} \Psi(\mathbf{X}, y, \mathbf{h}; \theta) = & \sum_{i=1}^{m} \sum_{k \in \mathcal{K}_i^1} \theta_k f_k^1(\mathbf{x}_i, h_i) + \\ & + \sum_{i=1}^{m} \sum_{k \in \mathcal{K}_i^1} \theta_k f_k^2(y, h_i) + \\ & + \sum_{(i,j) \in E} \sum_{k \in \mathcal{K}_{i,j}^2} \theta_k f_k^3(y, h_i, h_j) \end{array} \tag{9}$$

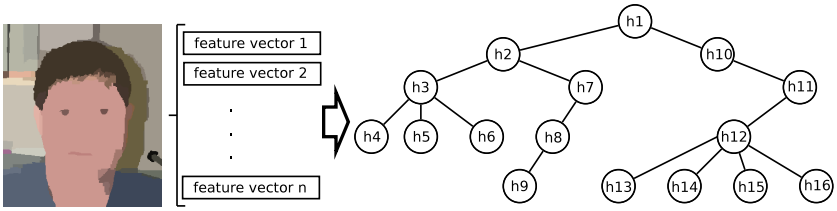

Figure 3: Overall CRF construction flow as explained in Section 4.2.

is used in our implementation, where $\mathcal{K}_i^1$ and $\mathcal{K}_{i,j}^2$ are the subsets of indices referring to the functions that depend on the i-th part label and on the i-th, j-th pair, respectively. It is explicitly remarked that every dependence on the particular position on the graph has been removed. The single functions have been chosen to be of three types (denoted by superscripts $1, 2, 3$ in Eq. (9)):

Type 1 functions encompass dependencies between features vector and the part label h. In our implementation there are $l \cdot |\mathcal{H}|$ functions $f_k^1(\mathbf{x}, h) = (\mathbf{e}_i \cdot \mathbf{x})\, \delta(h, h_j)$, where $i \in [1, l]$, $h_j \in \mathcal{H}$, $\mathbf{e}_i$ is the l-dimensional i-th unit vector and δ is the discrete Dirac delta.

Type 2 functions take into account dependencies between the part label h and the image category y: $f_k^2(y, h) = \delta(y, y_i)\, \delta(h, h_j)$ for each $y_i \in \mathcal{Y}$ and $h_j \in \mathcal{H}$. The number of these function is $|\mathcal{Y}| \cdot |\mathcal{H}|$.

Type 3 functions evaluate how much the presence of two neighbour part labels h^1 and h^2 relate to the image category y: $f_k^3(y, h^1, h^2) = \delta(y, y_i)\, \delta(h^1, h_j)\, \delta(h^2, h_k)$ and there are $|\mathcal{Y}| \cdot |\mathcal{H}|^2$ of these functions.

The classification of an image can be done calculating

$$y = \arg\max_{y' \in \mathcal{Y}} \left\{ p(y'|\mathbf{x}, \theta) \right\} = \arg\max_{y' \in \mathcal{Y}} \left\{ \sum_{\mathbf{h}} p(y', \mathbf{h}|\mathbf{x}, \theta) \right\} \quad . \tag{10}$$

The summations in Eq. (7), (10) would require an iteration over all the possible applications of each part label in $\mathcal{H}$ to each node. However they can be simplified due to the particular structure of the functions that have to be summed, that is, a product of functions that involve only a particular subset of the total number of variables. This function structure can be associated to a *factor graph* that can be solved via belief propagation [2, 9].

4.2.2 Parameters Learning

The choice of the parameters $\theta = \{\theta_1 \dots \theta_K\}$, where K is the number of feature functions (that should not be confused with the image features vectors $\mathbf{x}$) is accomplished through log-likelihood maximisation. The log-likelihood is:

$$L(\theta) = \sum_{i=1}^{N} \log \left(p(y_i|\mathbf{x}_i, \theta) \right) - \frac{\|\theta\|^2}{2\sigma^2} \quad , \tag{11}$$

where $p(y_i|\mathbf{x}_i, \theta)$ is obtained as in Eq. (10) and the second term is due to a Gaussian prior on L introduced in order to obtain a smoothed solution (to prevent the overfitting of the training set). In a model in which each parameter has a linear contribution on the feature functions, the gradient of the log-likelihood can be written in a form involving only the marginal probabilities $p(h^1|y, \mathbf{X}, \theta)$ and $p(h^1, h^2|y, \mathbf{X}, \theta)$ as shown in [12]. Such probabilities can be efficiently obtained from the factor graph introduced in Section 4.2.1.

The algorithm employed for the log-likelihood maximisation with the gradient information is the L-BFGS limited-memory quasi-Newton method for unconstrained optimisation [6]. This algorithm has been chosen due to the fast-convergence property of the quasi-Newton methods.

5. TESTS AND RESULTS

The system has been tested on a single-category classification task. The dataset used for the experiments is the Caltech 101 dataset [3], and the "faces" category was chosen to be discriminated against images randomly taken from all the other categories. The category set was thereby fixed to $\mathcal{Y} = \{\texttt{face}, \texttt{background}\}$.

The data sets were composed choosing 300 images from the faces category and other 300 images randomly from the other categories, then subdividing the set in three subsets equally dimensioned to obtain a training set, a test set and a validation set. Each subset was therefore composed of 100 "face" and 100 "background" images. The validation set was used to test the σ smoothing parameter in the log-likelihood evaluation, introduced in Eq. (11): the learning was performed with different values of σ and the model that performed the best based on the validation set was considered for the performance evaluation on the test set.

The images were preprocessed in order to extract the patches, as explained in Section 4.1. The number of obtained features for each image was 80 - 100, which is more than an order of magnitude less than in the reference model.

The model from Quattoni et al. [12] was chosen for results comparison. Two versions of the proposed model were tested, with the discretisation filter, introduced in Section 4.1, enabled and disabled respectively. The results obtained from the tests are shown in Table 1.

The first information arisen from the tests is related to the convergence problems of the algorithm. The L-BFGS algorithm failed finding the optimal solution for the given training set, and the best partial result on the log-likelihood maximisation had to be chosen in order to perform the accuracy evaluation. The model trained with features obtained without the use of the discretisation filter performed better. This behaviour can be explained with two arguments: the first is that the number of nodes increases as the regions' colours are not previously flattened. The second reason, that explains more generally the convergence problem, is that the local function's structure is too simple to embed the correct colour information, and the skin colour can not be adequately isolated.

On the other side, the improvement in terms of training speed of the framework is significant as expected.

Table 1: Comparison of performances between our model with the discretisation filter enabled ("our_{df}"), with the discretisation filter disabled ("*our*"), and the reference one ("*MIT*"). The number of iterations during the training, the relative training time, the sigma prior value and the classification accuracy are shown. †: the minor number of iterations is not due to settings but to the impossibility for the algorithm to find a better solution after that step.

Model	iterations	relative elapsed time	prior variance	accuracy
our_{df}	$68^{\dagger}$	0.04	10^4	77%
our	$79^{\dagger}$	0.14	1	83%
MIT	160	1	0.1	90%

6. CONCLUSIONS AND FUTURE WORK

The system has shown to work with overally acceptable results considering the basic information used for the classification task. The conclusions that can be drawn are:

- the CRF model is suitable to handle dependencies between coarse parts described via simple features;
- the reduction of the number of patches and of the dimensionality of the feature vectors leads to a notable increment in the system performances, making it useful in time-critical systems;
- the training of the CRF can be problematic when there is a lack of information possibly due to insufficient discriminative information, reduced dataset size, and oversimplification of the local function.

The CRF model can therefore be retained for further studies and improvements. The work can proceed with the aim to use the CRF to exploit dependencies from a small number of semantically meaningful image parts.

Research efforts should however be focused in finding representative features and feature functions to ease and stabilise the graph parameters learning while keeping the number of nodes small. Different maximisation algorithms for the log-likelihood can be tested as well. It is also possible to take advantage of the augmented speed that is due to the reduced number of nodes by introducing more complex features or graph structures.

7. ACKNOWLEDGEMENTS

The work leading to this paper was partially supported by the European Commission under contracts FP6-001765 aceMedia and EU COST Action 292. I would also like to acknowledge Joris Mooij from Radboud University Nijmegen, The Netherlands, for his support on the use of the *libDAI* library to perform the belief propagation on the CRF graph.

8. REFERENCES

[1] G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*, volume 147. Springer, second edition, 2006.

[2] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag New York Inc., 2006.

[3] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In *IEEE. CVPR 2004, Workshop on Generative-Model Based Vision*, 2004.

[4] S. Kumar and M. Hebert. Discriminative random fields: A discriminative framework for contextual interaction in classification. In *Proceedings of the 2003 IEEE International Conference on Computer Vision (ICCV '03)*, volume 2, pages 1150–1157, 2003.

[5] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th International Conf. on Machine Learning*, pages 282–289. Morgan Kaufmann, San Francisco, CA, 2001.

[6] D. C. Liu and J. Nocedal. On the limited memory bfgs method for large scale optimization. *Math. Program.*, 45(3):503–528, 1989.

[7] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004.

[8] L. Lucchese and S. Mitra. Colour segmentation based on separate anisotropic diffusion of chromatic and achromatic channels. *Vision, Image and Signal Processing, IEE Proceedings*, 148(3):141–150, June 2001.

[9] J. Mooij. http://www.mbfys.ru.nl/∼jorism/libdai/. libDAI - A free/open source C++ library for Discrete Approximate Inference methods.

[10] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(7):629–639, 1990.

[11] W. K. Pratt. *Digital Image Processing: PIKS Inside*. John Wiley & Sons, Inc., third edition, 2001.

[12] A. Quattoni, M. Collins, and T. Darrell. Conditional random fields for object recognition. In *Neural Information Processing Systems Vision*, 2004.

[13] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *Proceedings of European Conference Computer Vision (ECCV)*, 2006.

[14] J. Sivic, B. Russell, A. A. Efros, A. Zisserman, and B. Freeman. Discovering objects and their location in images. In *International Conference on Computer Vision (ICCV 2005)*, October 2005.

[15] C. Sutton and A. McCallum. An introduction to conditional random fields for relational learning. In L. Getoor and B. Taskar, editors, *Introduction to Statistical Relational Learning*. MIT Press, 2006. To appear.

[16] J. Weickert, B. Romeny, and M. Viergever. Efficient and reliable schemes for nonlinear diffusion filtering. *IEEE Transactions on Image Processing*, 7(3):398–410, March 1998.