OBJECT LOCALIZATION IN METRIC SPACES FOR VIDEO LINKING

Daniel Gatica-Perez [1] Ming-Ting Sun [2]

IDIAP–RR 03-09

JANUARY 2003

Dalle Molle Institute for Perceptual Artificial Intelligence • P.O.Box 592 • Martigny • Valais • Switzerland

phone +41 – 27 – 721 77 11
fax +41 – 27 – 721 77 12
e-mail secretariat@idiap.ch
internet http://www.idiap.ch

[1] IDIAP, Martigny, Switzerland
[2] University of Washington, Seattle, WA, USA

IDIAP Research Report 03-09

Object Localization in Metric Spaces for Video Linking

Daniel Gatica-Perez Ming-Ting Sun

January 2003

Abstract. While objects often constitute the desired level of access for browsing and retrieval in video databases, an inherent problem for on-line object definition is that of model construction from a few examples. In this paper, we present a probabilistic methodology to localize objects that appear across video segments, based on video structuring, object definition, and localization in the video structure. Localization is formulated as a problem of random sampling in a Metric Mixture Model framework, which allows for the joint modeling of a set of color appearance exemplars and their geometric transformations. To improve the efficiency of the sampling process, candidate configurations are drawn from a prior distribution using importance sampling, and evaluated using Bayes' rule. Experimental results on a database extracted from home videos depicting real objects (with variations of scale and pose) across video shots show the performance of the method.

1 Introduction

The design of tools for content-based video browsing and retrieval has a direct impact in digital libraries, professional and consumer content generation, and media delivery applications [24]. The first step in this direction has been the automatic extraction of video structure (summaries) from individual video clips. Summaries typically consist of a set of representative frames at different levels of a hierarchy (shots, scenes), which allow for browsing functions, limited to the image level [23], [18]. However, objects often constitute the desired level of access to a video database. In this view, the generation of links between video segments that contain objects of interest constitutes a valuable feature that complements the video summary representation [1], [16]. One problem of summaries is that the number of representative images in a real video can be quite large, thus complicating their usage. The capability of jumping backwards and forward in time to browse video based on user-defined objects provides more focused interaction.

Schemes for video object linking have been recently proposed [1], [11], [16]. The work in [1] automatically generates moving object links in videos, by first detecting foreground regions that do not conform a dominant (background) motion model, then extracting color-based features from such regions, and finally matching region features across key-frames extracted from shots. The work in [16] generates links for depth-layered regions in stereo video by a similar procedure, with the difference that depth segmentation is applied to extract foreground regions. In [11], a frontal face detector [17] was used to detect faces in videos, and a simple matching procedure was employed across all detected faces to generate face links. Most of these approaches have limitations arising from the ill-defined concept of "object", i.e., the image regions that should be detected and linked. In practice, most objects in videos (including people) are not motion-consistent, and have large variability of appearance and pose. Furthermore, automatic image segmentation continues to be an unsolved problem, in spite of progress [13].

By definition, video browsing is a task that requires human intervention. Furthermore, a truly interactive video object browsing system should provide the capability of defining objects on-the-fly. This point has generally been overlooked in previous work. In this paper, we propose to create video object links based on three steps: video structuring, object definition, and stochastic object localization in the video structure. Localizing objects is a fundamental problem in computer vision [21], [17], [12], [25], [2], [19], [26]. In brief, given a discriminative object representation, localization is a search problem in a configuration space, clearly demanding if the latter is large or continuous [17], [2]. An inherent problem for on-line object definition is that of model construction from a few examples, which imposes constraints on learning and inference schemes. We formulate the solution using a Metric Mixture model [22], which allows for the joint probabilistic modeling of exemplars and their geometric transformations in a space that has no vector structure. The probabilistic formulation is appealing as uncertainty is dealt with in a principled basis. Exemplars are object representations that can be readily extracted from raw data; in our case, they correspond to color image templates that represent an object of interest. After defining the configuration space, we address object localization by random sampling from a prior distribution [12], [20]. Candidate configurations are drawn using importance sampling [8], [10], which guides the search towards regions of the configuration space likely to contain the true configuration, thus avoiding exhaustive processing, and evaluated using Bayes' rule. To this purpose, we define an importance function based on parametric and non-parametric object color models. We illustrate the performance of our approach on a database of objects extracted from home videos that present variations of scale and pose.

The rest of the paper is organized as follows. Section 2 briefly describes the video structuring method. Section 3 presents in details the object localization algorithm. Section 4 describes the procedure for video link generation. Section 5 presents results and discusses the advantages and limitations of our approach. Section 6 provides some final remarks.

2 Video Structure Generation

A summarized four-level video structure (Fig. 6), consisting of representative frames extracted from video, scene, shot, and subshot levels, is generated by an algorithm based on domain-specific statistical models of visual similarity, temporal duration, and adjacency of video shots and scenes, and on the formulation of hierarchical clustering as sequential binary Bayesian classification [6]. After shot boundary detection, the algorithm treats each shot as a cluster, and successively evaluates the pair of clusters that yields the largest posterior odds L,

$$L = \frac{p(s|\mathcal{E}=1,\mathcal{I})\,\Pr(\mathcal{E}=1|\mathcal{I})}{p(s|\mathcal{E}=0,\mathcal{I})\,\Pr(\mathcal{E}=0|\mathcal{I})}, \tag{1}$$

where $\mathcal{E}$ is a binary variable that indicates whether the pair of segments correspond to the same scene or not, s denotes features extracted from a pair of segments, $\mathcal{I}$ denotes some knowledge about the world, $p(s|\mathcal{E},\mathcal{I})$ are the class-conditional pdfs of the observed features, and $\Pr(\mathcal{E}|\mathcal{I})$ is the prior of $\mathcal{E}$. A pair of segments will be merged only when $L \geq 1$, and the procedure will iterate until $L < 1$ for every pair of clusters. The algorithm generalizes previous time-constrained clustering algorithms [23]. The class-conditional distributions are represented by Gaussian mixture models (GMMs) of inter-segment visual similarity, temporal adjacency and duration. Visual similarity features are extracted from a set of representative frames, which are in turn extracted from each of the subshots (the latter approximately correspond to individual scene appearances). While the number of frames depends on shot appearance variation, the summary maintains a manageable number for object localization. A hidden set of frames is available for further search if necessary. Users specify objects of interest directly on the representative frames by drawing a bounding box around them (Fig. 1(a)). The process can be repeated in other frames where the object appears, to create a small set of color image templates, called exemplars in the following.

3 Probabilistic Object Localization in Metric Spaces

In pattern theory terms [9], an observed image $z \in \mathcal{Z}$ can be approximated by a template $x \in \mathcal{X}$ on which a continuous geometric transformation $t \in \mathcal{T}$ has been applied, $z \approx tx$. If the discrete set $\mathcal{X}$ represents an object model, $\mathcal{X} \times \mathcal{T}$ describes the object and its possible transformations. The representation is attractive: while $\mathcal{T}$ can model global motion, $\mathcal{X}$ can represent complex variations of shape, appearance, pose, and noise.

A probabilistic formalization of this approach was developed in [5], and generalized in [22] for non-vector exemplars [7]. Exemplars are convenient low-level object representations (color or edge image templates) because they can be extracted relatively easily from images, and then used to define object models, without resorting to elaborate intermediate representations. However, basic operations for probabilistic modeling, like averaging, are not defined in a space that has no vector structure [14]. For object tracking, the work in [22] described in a principled way how probabilistic mixture models can be defined and learned from exemplars in a metric space [1]. The core concepts are the use of exemplars as "centers" of a mixture model, and the definition of metrics (or distance functions) that are good to compare exemplars in the space in which they live, thus overcoming models that disregard statistical correlation between pixels [5].

We propose to use a similar formulation for object localization. In our case, $\mathcal{X}$ consists of the set of user-defined color image templates $\tilde{x}_k$ that model object appearance, $\mathcal{X} = \{\tilde{x}_k, k = 1, ..., K\}$, equipped with a distance function ρ. Additionally, the transformation space $\mathcal{T}$ is defined as a subspace of the affine transformations that models translation and scaling, which is useful to locate targets. Elements of the exemplar-transformation space will be denoted by the pair $X = (k, t)$.

[1] A metric space consists of a set R and a real-valued function ρ called *metric* that satisfies (1) $\rho(r,s) \geq 0$, (2) $\rho(r,s) = 0$ iff $r = s$, (3) $\rho(r,s) = \rho(s,r)$, and (4) $\rho(r,s) \leq \rho(r,q) + \rho(q,s)$, $\forall r,s,q \in R$ [14]. A *distance function* satisfies only conditions 1 and 2. It was shown in [22] that the theory applies to distance functions as well.

3.1 Formulation of the localization problem

In similar fashion to [12], [20], we formulate object localization using Bayesian theory and stochastic simulation. Given a prior distribution on the possible object configurations $p(X)$, an observed image z, and an observation likelihood $p(z|X)$, the posterior is expressed by Bayes' rule as

$$p(X|z) \propto p(z|X)p(X). \tag{2}$$

A probabilistic generative model for this formulation, assuming independence between t and k, is shown in Fig. 1. The model has a simple interpretation: an observed image z is the result of a process in which a transformation t and an exemplar index k are independently chosen with probability $p(t)$ and $p(k)$, respectively, followed by drawing an exemplar from $p(\tilde{x}|k)$ (that in general could model a noise process in exemplar space before transformations). The image z is then drawn from $p(z|\tilde{x}_k, t)$. In the simplest formulation, $p(\tilde{x}|k) = \delta(\tilde{x} - \tilde{x}_k)$, where $\delta(\tilde{x} - \tilde{x}_k)$ denotes a Dirac delta located at $\tilde{x}_k$, so $p(z|\tilde{x}_k, t) = p(z|k, t)$.

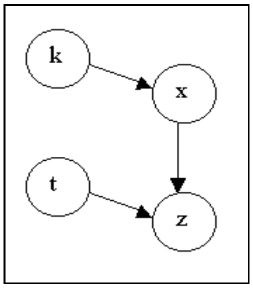


Figure 1: Graphical model for object localization.

There are well-known Monte Carlo discrete representations for distributions, discussed elsewhere [4]. In brief, a distribution can be approximated by a set of weighted samples or particles,

$$\{(X^{(i)}, \pi^{(i)}), i = 1, ..., N\},$$

where $X^{(i)}$ and $\pi^{(i)}$ denote the i-th sample and its weight, respectively, and the weights are given by $\pi^{(i)} = p(z|X = X^{(i)})$. After normalizing the weights, the point-mass approximation is

$$\hat{p}_N(X|z) = \sum_{i=1}^{N} \pi^{(i)} \delta(X - X^{(i)}),$$

where the Dirac delta is located at $X^{(i)}$. From the above equation, statistics on X can be estimated. In a vector space [12], [20], moments of the posterior can be readily computed [4],

$$E(f(X)) \approx \sum_{i=1}^{N} \pi^{(i)} f(X^{(i)}).$$

In contrast, averages are not defined in a space that has no vector structure. However, peaks in the posterior still provide evidence of object location. Therefore, our approach for localization draws a set of random proposals S from the prior $p(X)$, evaluates the observation likelihood at each proposal (extracting image measurements, and indeed quantifying discrepancy between the prior and the true posterior distribution), and chooses the configuration that maximizes the posterior in the sample set,

$$X^* = \arg \max_{X^{(i)} \in S} \hat{p}_N(X|z). \tag{3}$$

This formulation requires the specification of the distributions and a decision rule to decide whether the object is present in the scene.

3.2 Modeling the observation likelihood

We propose to model the observation likelihood by a *metric exponential* [22],

$$p(z|X) = p(z|k,t) \propto \frac{1}{\mathbf{Z}} e^{-\lambda\rho(z,t\tilde{x}_k)}, \tag{4}$$

where $t\tilde{x}_k$ corresponds to a transformed exemplar, $\mathbf{Z}$ is a normalization constant (also called partition functions), and λ is a parameter that has to be estimated from data. As t and k are independent, the likelihood on transformations is a mixture of metric exponentials,

$$p(z|t) = \sum_{k=1}^{K} p(z,k|t) = \sum_{k=1}^{K} p(k)p(z|k,t), \tag{5}$$

whose centers are the transformed exemplars $t\tilde{x}_k$, and whose weights are given by the exemplar prior $p(k)$. Obviously, this interpretation only holds when the dimension d_k of the space defined on z for all K components of the mixture are identical (otherwise, the sum in Eq. 5 cannot be done). This condition might not be true in general, but the expression for the joint distribution (Eq. 4) is still valid for all cases, and it is used in this paper. We further assume a quadratic form as a reasonable noise model, when ρ is the distance function based on the Bhattacharyya coefficient [3]. Such function has proven to be useful to compare object and target color distributions [2]. In that case, the exponential parameter and the normalization constant can be approximated by $\lambda \approx \frac{1}{2\sigma^2}$ and $\mathbf{Z} \propto \sigma^d$, where σ is a "variance" parameter [3] that measures the spread of the metric exponential "around" its center, and d is a measure of the "effective" dimensionality of the unknown exemplar space. The chosen distance function is defined by

$$\rho(z,t\tilde{x}_k) = (1 - d_{BT}(f(z), f(t\tilde{x}_k)))^{1/2}, \tag{6}$$

where $f(t\tilde{x}_k)$ denotes a 4-D normalized histogram of the transformed exemplar that includes 3-band color plus the relative position of each pixel in the template as components, and $f(z)$ denotes the observed 4-D image histogram computed over the support of $t\tilde{x}_k$. The relative position component models basic spatial structure. Additionally, the Bhattacharyya coefficient is defined by $d_{BT} = \sum(f(z)f(t\tilde{x}_k))^{1/2}$, the sum running over all bins in the histogram. Except for quantization effects, the normalized histogram is translation- and scale- invariant [21], unlike other representations, like coocurrence histograms [2], which are not scale-invariant, and therefore allows us to approximate $f(t\tilde{x}_k)$ by $f(\tilde{x}_k)$ in the distance function computation. The relative position component used to model spatial structure consists of three bins, dividing the template in three vertical subblocks of identical size.

Parameters can be estimated as described in [22]. Given a set of user-defined exemplars, and a training set of (additional) color object templates $\mathcal{Z} = \{z_v\}$, each of the latter is first assigned to one exemplar by minimizing the distance function $\rho(z_v, t\tilde{x}_k), \forall t \in \mathcal{T}, \tilde{x}_k \in \mathcal{X}$. Then, the set of resulting distances $\{\rho_v(z_v)\}$ is assumed to be $\sigma^2\chi^2$ distributed, and parameters of the model are estimated from sample moments,

$$\bar{\rho}_k = \frac{1}{N_k} \sum_{z_v \in C_k} \rho_v(z_v), \qquad \bar{\rho_k^2} = \frac{1}{N_k} \sum_{z_v \in C_k} \rho_v^2(z_v),$$

where C_k denotes the cluster represented by exemplar $\tilde{x}_k$, which leads to

$$d_k = 2\frac{\bar{\rho}_k^2}{\bar{\rho_k^2} - \bar{\rho}_k^2}, \qquad \sigma_k = (\bar{\rho}_k/d_k)^{1/2}.$$

[2] ρ_{BT} is a true metric in functional space, but not in exemplar space.

[3] A true variance for certain distance functions [22].

From the above equations, it can be seen that the dimensionality of the space for each component (each individual exemplar) is related to the distance functions computed in the training set. For tracking purposes, exemplars were clustered to reduce the model complexity of the observation likelihood, and parameters were estimated from hundreds of examples [22]. However, for video object linking, users usually specify one or a handful of exemplars, so it is not possible to perform on-line clustering and estimation from such amount of data. Instead, we have estimated the parameters for several objects on training videos, and used the same parameters for all new cases (see Section 5). Additionally, each of the user-specified exemplars is treated as a center in the Metric Mixture. More satisfactory solutions are under study.

3.3 Importance sampling from the prior

The prior distribution $p(X)$ encodes the knowledge about object location. As stated before, exemplar indices and geometric transformations are assumed to be independent, so $p(X) = p(k,t) = p(k)p(t)$. The most general assumption is a uniform distribution $u(\cdot)$ on both exemplar index and geometric transformations (in the latter case, over a closed interval). However, knowledge about possible locations at each representative frame can be extracted using object features, like color or texture, and could be useful to guide the random search. This is properly modeled through the use of importance sampling [8], [10], a well-known method to improve the efficiency of simulation methods, and useful when such additional knowledge can be expressed by a (normalized) importance function $g(X)$ that emphasizes the most "informative" regions of the configuration space. The technique first draws random samples $X^{(i)}$ from $g(X)$ rather than from $p(X)$, which concentrates particles in better proposal regions, and then introduces a correction mechanism in order to keep the particle set as a faithful representation of the original distribution. Such correction takes the form of an importance ratio factor defined by $p(X = X^{(i)})/g(X = X^{(i)})$, and applied on the particle weights $\pi^{(i)}$ [10], [4]. In our work, we keep the assumption of uniformity on exemplars, $p(k) = u(k)$, and use importance sampling to draw samples from the transformation distribution $p(t)$.

3.4 Constructing the importance function

We use a parameteric color model of each exemplar to generate candidate configurations in each of the representative frames in which the object is searched for. Let y represent an observed color vector for a given pixel. Given a single foreground object, the distribution of y is a mixture, $p(y|\Theta) = \sum_{i \in \{F,B\}} p(O_i)p(y|O_i,\theta_i)$, where F and B stand for foreground and background, $p(O_i)$ is the prior probability of pixel of color y of belonging to object O_i, and $p(y|O_i,\theta_i)$ is the conditional pdf of observations given object O_i, represented by a GMM. In absence of prior knowledge ($p(O_F) = p(O_B)$), the optimal classification into foreground/background is obtained by comparing the likelihood ratio $\frac{p(y|O_F,\theta_F)}{p(y|O_B,\theta_B)}$ to 1.

Color models are on-line estimated for each exemplar using the Expectation-Maximization (EM) algorithm [8]. Then, for each searched image, a binary image I_k^b is built based on pixel classification, followed by morphological processing, in order to generate blobs whose colors match the object model. As the background color distribution is likely to change from shot to shot (possibly rendering low values for $p(y|O_B,\theta_B)$) probabilities are thresholded to ensure that the generated blobs truly correspond to object colors. Finally, a blob image is obtained by computing the maximum of the binary images obtained for each exemplar, $I^g = \vee_{k=1}^{K} I_k^b$. An example is shown in Fig. 2, for one exemplar.

Recall that the transformation space $\mathcal{T}$ has been chosen to include translation (o) and scaling (s), so any $t \in \mathcal{T}$ can be denoted by $t = (o,s)$. Assuming independence and a uniform distribution for the scaling parameter, the importance function is $g(t) = g(o,s) = g(o)u(s)$. To specify a functional form for the translation parameters $g(o)$, we use the binary image I^g. We define $g(o)$ as a GMM,

$$g(o) = g(o|\Phi) = \sum_{i=1}^{C} p(c_i)p(o|c_i,\phi_i), \qquad (7)$$

where c_i denotes each of the C connected components of I^g. The parameters ϕ_i correspond to the mean and 2-D covariance matrix of the pixels in each component. Furthermore, the prior distribution $p(c_i)$, which defines the relative contribution of each blob to the mixture, is determined from two features: the blob size, and its maximum color similarity (i.e. the minimum distance ρ_{BT}) to the exemplars that define the object model,

$$p(c_i) = \sum_{j \in \{size, color\}} p(w_j) p(c_i | w_j), \tag{8}$$

The distributions $p(c_i|w_{size})$ and $p(c_i|w_{color})$ are directly estimated from data. Finally, the prior $p(w_j)$ is assumed uniform. Random sampling will draw more configurations from large blobs whose color distribution better matches the object (Fig. 2(c)).

3.5 Object detection/absence

The described method outputs both the geometric transformation and the exemplar that best match the object model for each representative frame. Object absence is decided based on thresholding of the particle weights. The threshold is empirically determined from a set of positive and negative examples.

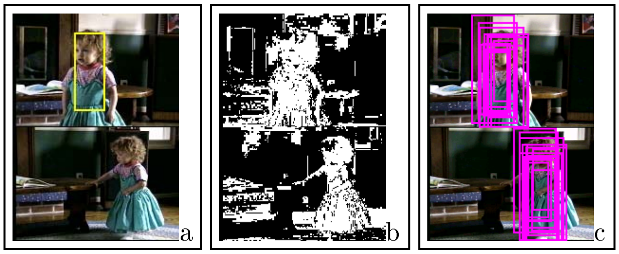


Figure 2: (a) Frames from *Girl* video (the template defined inside the yellow line constitutes the exemplar X_1). (b) Binary image after pixel classification. (c) Importance sampling. The displayed samples have the largest weights (Eq. 3). Only the transformed template contours are shown.

4 Video link generation

Links are constructed based on object detection/absence for each subshot. The leaves in the video structure that contain the object are highlighted, as shown in Fig. 6. Alternatively, video object links could be required only at higher levels of the hierarchy (shot, cluster). In that case, the algorithm is applied until it detects an object, and then moves to the next shot or cluster.

5 Results and discussion

5.1 Dataset and performance measures

Performance evaluation of user-defined object localization algorithms in videos poses several issues to consider. The problem is different from traditional object localization. On one hand, there is no strong prior knowledge of the target objects (as opposed to the “face/car” typical problem [17], [19]). On the other hand, objects have different degrees of complexity and variability and as stated earlier, we cannot expect to have a large amount of training data for each object. Furthermore, while raw video represents a potentially enormous amount of test data (as an object of interest might be present in thousands of adjacent frames in the same shot), the number of object instances in a video summary will be small in general. To our knowledge, there are no standard databases for video object localization in real scenes, which could be the video equivalent to commonly used image databases (e.g. [15]).

Based on these factors, we constructed an initial dataset consisting of ten objects extracted from six 20-minute videos from a real home video database [6]. All videos were filmed with hand-held cameras, and depict typical scenes (outdoor, indoor, children playing, family and school parties) with distinct degrees of quality. Each of the objects appeared in a variable number of shots, with natural changes of pose, scale, and illumination, and partial occlusion. The database includes two parts. For the first one, we manually extracted video segments depicting each of the objects, and then sample 100 frames from each set of segments. Some images featuring the objects are shown in Fig. 3 (letters a-j correspond to object O1-O10). Additionally, we extracted 40 extra frames for three of the ten objects, and labeled them manually for training purposes (in the following, training objects are labeled as O1-O3). For the second part, we generated a random sample of 500 images extracted from ten 20-minute videos that do not contain any of the objects. The final image set thus consists of 120 training images, and 1500 test images. Finally, we use several excerpts of the original video clips to run the localization algorithm on all of their frames (in this case, we did not measure performance).

After framing object localization as a retrieval problem, results were evaluated by computing precision as a function of recall. Precision is the number of retrieved images that contain the queried object ("true-object" images) relative to the total number of retrieved images. Recall is the number of retrieved "true-object" images relative to the total number of "true-object" images in the database.

5.2 Results

The results presented here correspond to the one-exemplar case. The RGB space was used to compute the normalized histograms ($8 \times 8 \times 8$ bins) and the parametric models. Table 1 shows the estimated parameters for the metric mixture model, for each of the three objects for which training images were extracted, and for all the distance measurements combined as if they had come from the same object. We observe that the variation in the parameters is quite significant from object to object due to individual object appeareance variability. This variation is confirmed when looking at the histograms of the distance measure based on Bhattacharyya-coefficient for the objects in the training set. For all the experiments, we used the set of parameters described in the last row of Table 1.

Object number	d	σ	λ
O1 (Girl)	24.06	0.15	22.22
O2 (Wedding Bride)	9.05	0.21	11.57
O3 (Wedding Man)	16.60	0.20	11.91
All objects	14.34	0.16	18.17

Table 1: Estimated parameters for metric exponential model.

Precision and recall results are presented in Fig. 4, for a fixed value of 300 random samples drawn from the prior, and a range for the scale parameter of [0.5, 2]. While the performance of the method varies across objects, good values for recall (79%) and precision (82.5%) were obtained for the entire database, averaging over all objects. The results are not surprising given the objects characteristics, but they suggest the performance of the method in locating objects inside and across shots, when the object model is discriminative enough such that appearance variations can be handled by the color representation, even with changes of background. Some localization results for each object are shown in Fig. 3(a-j). For comparison, Fig. 3(k) shows the best ten results obtained with exhaustive search for O1, with translation quantized by a factor of 4 in each direction, and scaling quantized to 10 levels (13200 configurations). Additionally, the results of running the localization algorithm on excerpts of video sequences should be fully appreciated by looking directly at the videos, available on a website that accompanies this paper [4]. The computational complexity is dependent on object size. After color

[4] www.idiap.ch/˜gatica/mmir/object_localization.html

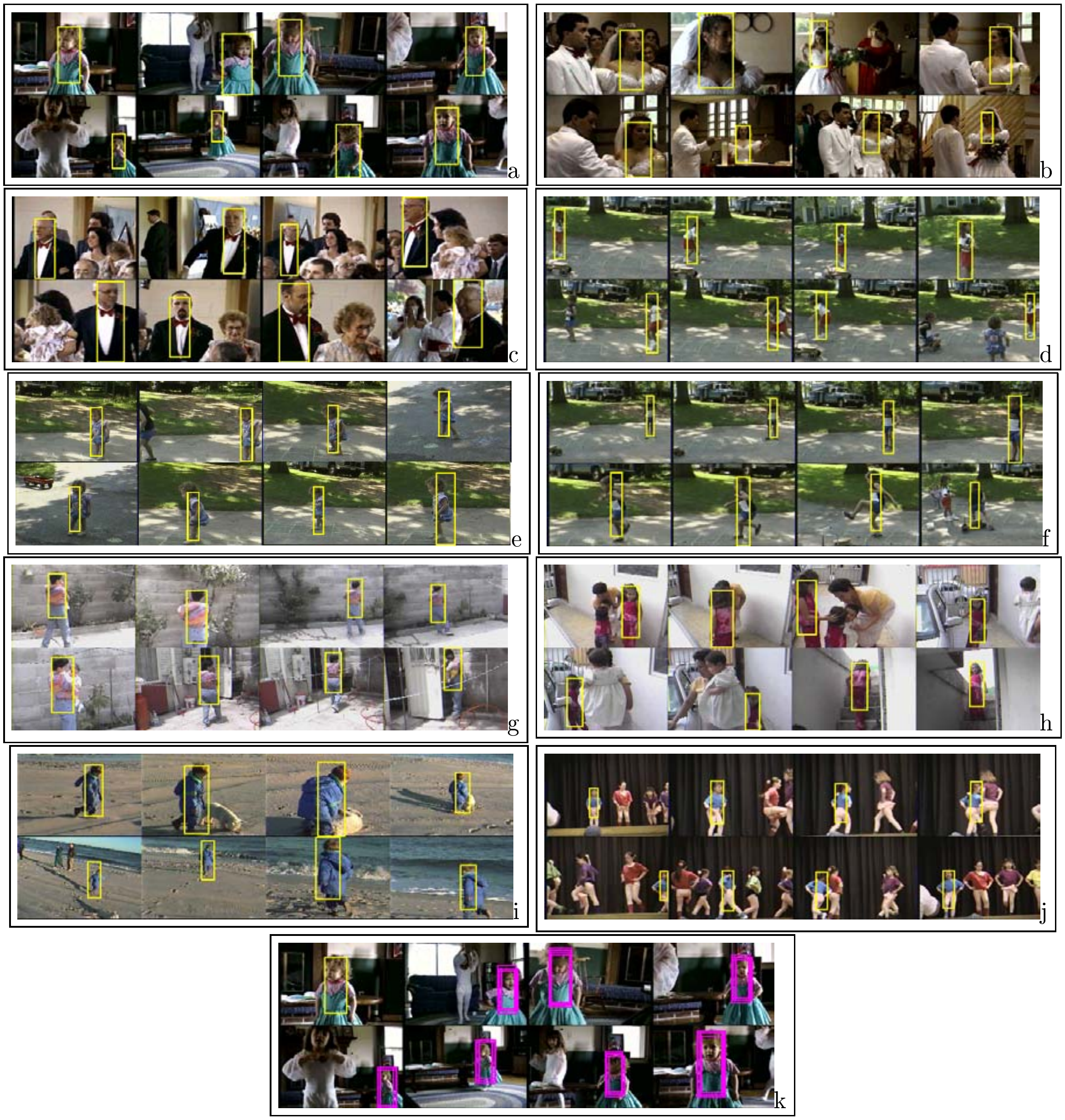


Figure 3: Object localization by (a-j) sampling (only the best configuration is shown). In all cases, one exemplar is specified in the first frame; (k) localization by exhaustive search for one case.

model estimation, it takes approx. 0.5 s. to process 300 samples per QSIF image, on a Pentium III, 600 MHz PC. This figure could be significantly improved by code optimization and multi-resolution processing. As a byproduct, the method could be used to initialize an stochastic tracker [10] for further video analysis inside each shot.

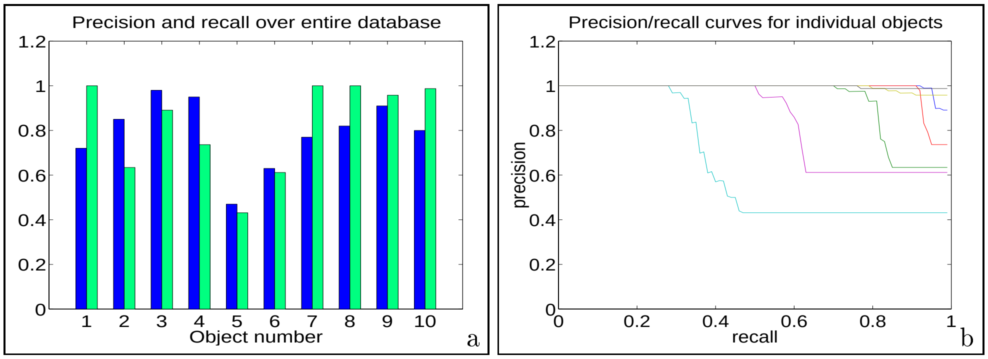


Figure 4: (a) Recall (blue bar) and precision (green bar) for each object in the database. (b) Precision as a function of recall (a different color line for each object.)

5.3 Limitations

While the obtained results are encouraging, the algorithm is limited by three factors: (i) the assumptions about admisible object shape (bounding boxes) and geometric transformations (rigid, and limited to three parameters), (ii) the discrimination that can be obtained with color histograms, and (iii) the quality of the importance function. On one hand, the observation likelihoods generated for color templates represented by histograms object tend to be broad. The introduction of spatial structure improves the results compared to color-only histograms, but still generates likelihoods that can be unspecific. Two objects with similar color/spatial distribution are indistinguishable. This can be seen in Fig. 5, where some retrieval errors are shown for a few objects in the database. On the other hand, an importance function that filters out most of the scene while retaining the approximate object position is required for the sampling mechanism to succeed. The algorithm will fail whenever such function cannot be generated.

6 Conclusion and future work

We presented a methodology to create video object links based on video structuring and a probabilistic formulation of object localization, a process of random search in a configuration space of exemplars and geometric transformations that considerably reduces computational complexity compared to exhaustive search, while keeping good localization features. Results of good quality for object video browsing in real videos have been obtained, but there is room for improvement. Several issues are currently under analysis, including the use of illumination-invariant color models, better modeling of spatial structure, and the definition of a decision mechanism based on probabilistic models. Finally, devising mechanisms for parameter estimation in the metric mixture model requires further study.

7 Acknowledgements

The authors thank the Swiss National Science Foundation for supporting this work through the National Centre of Competence in Research (NCCR) on "Interactive Multimodal Information Management (IM2)", the Eastman Kodak Company for providing the Home Video Database, and Jean-Marc Odobez for discussions.

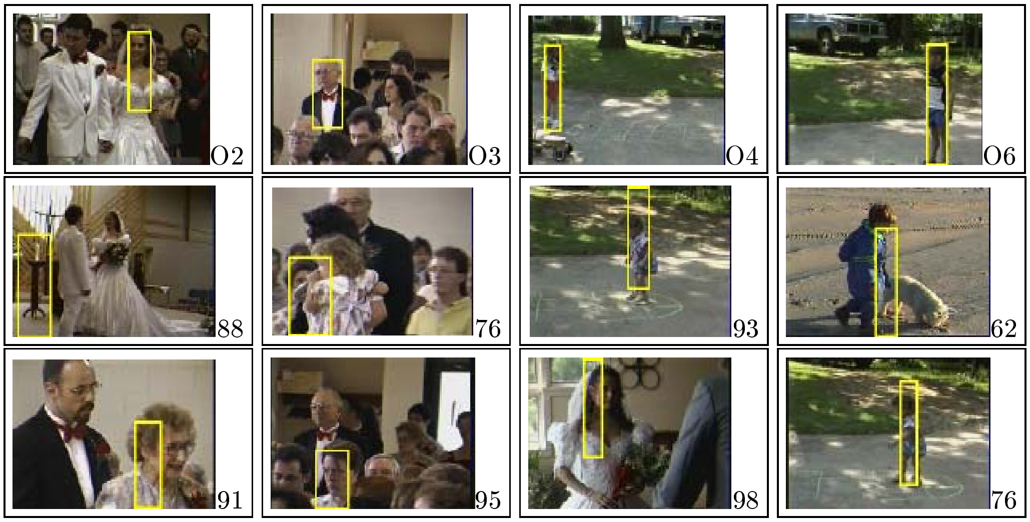
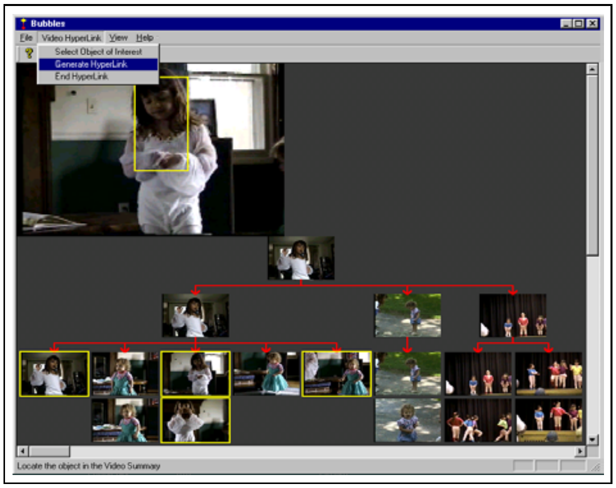


Figure 5: Typical errors. The objects displayed in the top row were used as queries. Each column corresponds to results obtained for objects O2, O3, O4, and O6, respectively. For each case, the object localization algorithm was run on the entire database (1500 images), and the best 100 results (according to Eq. 3) were retrieved. The middle and bottom rows show retrieved images with localization errors for each object. The number next to each image indicates its relative order in the list of best matches (the lower the number, the better the match).



Locate the object in the Video Summary

Figure 6: Video tree structure and object links. The root node corresponds to the video sequence, the middle nodes to the scenes, the leaf columns represent the shots, and individual leaves are frames extracted from subshots. Frames where the object has been located are highlighted.

References

[1] P. Bouthemy, Y. Dufournaud, R. Fablet, R. Mohr, S. Peleg, and A. Zomet, “Video Hyper-links Creation for Content-Based Browsing and Navigation,” in *Proc. CBMI*, Oct. 1999.

[2] P. Chang and J. Krumm, “Object Recognition with Color Coocurrence Histograms,” in *Proc. IEEE CVPR*, Fort Collins, CO, June 1999.

[3] D. Comaniciu, V. Ramesh, and P. Meer, “Real-Time Tracking of Non-Rigid Objects using Mean Shift,” in *Proc. IEEE CVPR.*, Hilton Head Island, S.C., June 2000.

[4] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*, Springer-Verlag , 2001.

[5] B. Frey and N. Jojic, “Learning graphical models of images, videos and their spatial transformations,” in *Proc. UAI*, 2000.

[6] D. Gatica-Perez, M.-T. Sun, and A. Loui, “Consumer Video Structuring by Probabilistic Merging of Video Segments,” in *Proc. IEEE ICME*, Tokyo, Aug. 2001.

[7] D. Gavrila and V. Philomin, “Real-time object detection for smart vehicles,” in *Proc. IEEE ICCV*, 1999.

[8] A. Gelman, J. B. Carlin, H. S. Stern and D. B. Rubin, *Bayesian Data Analysis.* Chapman and Hall, 1995.

[9] U. Grenander, *Lectures in Pattern Theory.* Springer, 1981.

[10] M. Isard, and A. Blake, “ICondensation: unifying low-level and high-level tracking in a stochastic framework,” in *Proc. ECCV*, 1998.

[11] W.Y. Ma and H.J. Zhang, “An Indexing and Browsing System for Home Video,” in *Proc. EUSIPCO.* Patras, 2000.

[12] J. MacCormick and A. Blake, “A probabilistic contour discriminant for object localisation,” in *Proc. IEEE ICCV*, pp. 390-395, Bombay, Jan. 1998.

[13] M. Meila and J. Shi, “A random walks view of spectral segmentation,” in *Proc. AISTATS*, 2001.

[14] A. W. Naylor and G. R. Sell, *Linear Operator Theory in Engineering and Science*. Springer-Verlag, 1982.

[15] S. A. Nene, S. K. Nayar and H. Murase, “Columbia Object Image Library (COIL-100),” Columbia University Technical Report CUCS-006-96, Feb. 1996.

[16] K. Ntalianis, A. Doulamis, N. Doulamis, and S. Kollias, “Non-Sequential Video Structuring Based on Video Object Linking,” in *Proc. IEEE ICIP*, Thessaloniki, October 2001.

[17] H. Rowley, S. Baluja, and T. Kanade, “Human Face Detection in Visual Scenes,” TR-CMU-CS-95-158R, Nov. 1995.

[18] Y. Rui and T. Huang, “A Unified Framework for Video Browsing and Retrieval,” in Alan Bovik, Ed., *Image and Video Processing Handbook*, Academic Press, 2000.

[19] H. Schneiderman and T. Kanade, “Probabilistic Modeling of Local Appearance and Spatial Relationships for Object Recognition,” in *Proc. IEEE Int. Conf. Pat. Rec. and Computer Vision*, Santa Barbara, 1998.

[20] J. Sullivan, A. Blake, M. Isard and J. MacCormick, “Object Localization by Bayesian Correlation,” in *Proc. IEEE Int. Conf. Computer Vision*, pp. 1068-1075, 1999.

[21] M.J. Swain and D. Ballard, “Color Indexing,” *Int. J. of Comp. Vis.*, Vol. 7, pp. 11-32, 1991.

[22] K. Toyama and A. Blake, “Probabilistic Tracking in a Metric Space,” in *Proc. IEEE ICCV*, Vancouver, Jul. 2001.

[23] M. Yeung, B.L. Yeo, and B. Liu, “Segmentation of Video by Clustering and Graph Analysis,” *Computer Vision and Image Understanding*, Vol. 71, No. 1, pp. 94-109, July 1998.

[24] H.J. Zhang, “Content-based Video Browsing and Retrieval,” in *Handbook of Multimedia Computing*, CRC Press, 1999.

[25] Y. Zhong and A. K. Jain, “Object Localization Using Color, Texture and Shape,” in *Proc. Workshop on EMMCVPR*, Venice, pp. 279-294, May 1997.

[26] X. S. Zhou, B. Moghaddam, T. S. Huang, “ICA-based Probabilistic Local Appearance Models,” in *Proc. IEEE ICIP*, Thessaloniki, October 2001.