Spatio-temporal constraints for view-based description of 3D objects in videos

Elisabetta Delponte, Nicoletta Noceti, Francesca Odone, Alessandro Verri

1 Introduction

The problem of describing 3D objects from visual information without explicitly computing their 3D structure has been extensively studied in the last decade. Among the better established methods it is worth mentioning multiple-view approaches and local approaches. *XXX aggiungere referenze* There have also been attempts to combining the two, usually merging local features extracted from different images, but rarely the fact that a meaningful image feature was persistent at different viewing angles has been conveniently studied. In this paper we explore in depth the interlink between temporally dense view-based object recognition and sparse image representations with local keypoints. The temporal component is an add on that allows us to extract information which is distinctive of a given object in a given view-point range.

Our work is motivated by the fact that an image sequence does not just carry multiple instances of the same scene, but also information on how the appearance of objects evolves when the observation point changes smoothly. Since in many applications image sequences are available and often under exploited, our aim is to fill this gap.

We believe that, in the appropriate application setting, observing an object from slightly different viewpoints, and looking for features that are distinctive in space, stable and smooth in time, can greatly help object recognition. Moreover biological vision systems gather information by means of motion (motion of the pupil and motion of the head). This information includes important cues for depth perception and object recognition. To the purpose of recognition, fixating an unfamiliar object for a few seconds and possibly observing it from slightly different viewpoints is a common practice in human experience. Most optical 2D illusions of 3D

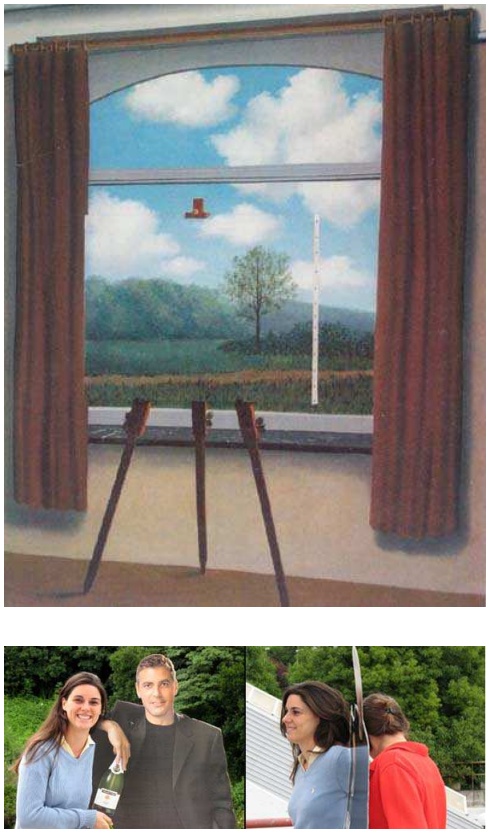

Figure 1: Playing with 2D and 3D scenes: (top) a famous Magritte's masterpiece: the difference between the 3D world outside the window and the 2D painting is suggested by the artist with a careful choice of the observer's viewpoint, that reveals the canvas side; (bottom) 2D objects in 3D world can be spotted only from the right angle.

shapes would be disambiguated if multiple views of the 3D object were available (see Figure 1). Sometimes it suffices to move the observation point slightly to get an entirely new perception of the observed scene. It is known that the cerebral cortex uses spatio-temporal continuity to build invariant representations w.r.t. scale, view-point, and position variation of 3D objects [SPRP06].

The contributions of our work are manifold: firstly we propose an appearance-based recognition method that exploits temporal coherence *both in training and test*, thus the method fits naturally in the video analysis framework. In spite of the redundancy of the data that we use, the descriptions we obtain are compact since they only keep information which is very distinctive across time. A second important contribution is the matching procedure: simple nearest

neighbor is strengthened with a strategy that favors matching between groups of features which are coeval, i.e. features belonging to similar fields of view, and which are spatially neighbours. In other words, we discard those features which are spatially and temporally isolated. Finally we argue that, modeling the scene by means of a continuos set of images or a video sequence, we may

- capture the idea of invariant representation to view-point changes and translation which is typical of biological vision systems
- increase the recognition confidence by observing an object for an amount of time which is proportional to the object peculiarity, its difficulty and the divergence of the acquisition conditions with respect to the training stage.

We report very good results on test sequences of increasing difficulty, in the presence of clutter, illumination and scene changes, occlusions, similar objects. *Ampliare questa descrizione breve degli esperimenti riportati nell'articolo distinguendo le due fasi.*

1.1 Related work

In the following, we give a brief hystoric overview of current approaches to object recognition: from first view-based techniques to recent developement of methods based on the use of codebooks. Here we focus mainly on those techinques based on the use of temporal information and local descriptors.

There are several reasons motivating the introduction of view-based approaches to object recognition, among which we recall biological inspiration. The research in neuroscience has shown that, for recognising objects, primates use simple features which are invariant to changes in scale, location and illumination [Tan97]. *Qui bisogna vedere se si puo' citare l'ultimo articolo di Poggio, quello con Serre che e' un po' piu' recente rispetto a questo.* The approach proposed by Edelman, Intrator and Poggio in [EIP] makes use of a model of biological vision based on complex neurons in primary visual cortex which respond to a gradient at a particular orientation but allows for shift over a small receptive field rather than being precisely localise. They hypothesised that the function of these complex neurons was to allow for matching and recognition of 3D objects from a range of viewpoints. Over the last decade, a number of physiological

studies in nonhuman primates have established several basic facts about the cortical mechanisms of recognition. A particularly important aspect of visual system is the back-projection, abundantly present between almost all of the areas in the visual cortex. In [SOP06] the authors propose a feedforward architecture based on the simple-to-complex cell hierarchy and on the use of learning at different stages of the model. They test the model with a categorization problem which gives results comparable with human capabilities of recognition.

Forse si puo' eliminare da qui XXXXX Among view-based techniques, it is worth mentioning 2D morphable models [PV92, BSP93, VP97, JP98] and view-based active appearance models [ECT99, CWT00], which use a selection of 2D views for the purpose of modelling a 3D complex object. The idea behind both methods (which achieve comparable results in two rather different ways) is that of synthesising a model for each object or class of objects, and then matching it with a novel image to check for consistency. Both morphable models and active appearance models are firmly based on registering a selection of images belonging to the same object or the same class, and compute linear combination of the examples in terms of their 2D shape and their texture, which represent a model of the object. *Fino a qui XXXXX*

The popularity of local approaches [Low99, CDFB04, LMS06, RLSP06, FTG06, TFL$^+$06] is due to the fact that, unlike global methods [MN95, PV98], they produce relatively compact descriptions of the image content and do not suffer from the presence of cluttered background, scene variations and occlusions. Also, by means of appropriate descriptions, they can be matched effectively, even in the presence of illumination changes and scale variation.

A number of 2D local keypoints detectors and descriptors have been proposed in the literature, and comparative evaluations have been proposed [MS05, MP05]. Of particular interest to our study is the work carried out in [MP05]: the authors assess feature detectors and descriptors for 3D objects, pointing out how, on a generic 3D object, features are partly due to textures and partly generated by the object shape (edges, folds, ...). The latter are more unstable to large view-point changes, therefore Moreels and Perona conclude that an affine-invariant detector is more appropriate. In our case, since we consider smooth variations on an image sequence, we rely on simplest detectors. As for the description we resort to SIFT [Low04], considered the most reliable in a variety of situations [MS05, MP05]. Despite these studies, it is not yet clear if one keypoint is best suited for object recognition or for other tasks [Lei04]. Our experience lead

us to the conclusion that the exact choice of keypoint detectors is not crucial: each keypoint can be more suitable to describe a given image in relation with the qualities of the image itself. Instead it is worth to remember the importance of robustness of descriptors to image variations such as scale, illumination and orientation changes. On this respect, there is a general belief that SIFT descriptors are a good choice.

Local features have also been already used to recognize 3D objects from multiple views, see [RLSP06, FTG06] or [BL05], where features are also used to build an explicit 3D model of the object as well as recognizing it. Recently multiple view object categorization has been addressed in [TFL$^+$06]. A novel and interesting approach to multiple view object categorization is described in [SFF07]: the authors propose a novel 3D model of an object class by encoding both the appearance and 3D geometric shape information.

Ci si potrebbe mettere una citazione a oliver sacks... o e' da sboroni? The main problem with local methods is that, while observing minute details, the overall object's appearance may be lost. Also, small details are more likely to be common to many different objects (this feature has actually been exploited in [TMF06] to design efficient multi-class systems). For this reason local information is usually summarized in global descriptions of the object, for instance in codebooks [CDFB04, LMS06]. Alternatively, closeness constraints can be used to increase the quality of matches [FTG06]. Moreover there have been some attempts to introduce information about global appearance and context of the image [TMF06, FTG06, RVG$^+$07].

Temporal continuity may help to cluster related keypoints observed at different view-points, in the case a dense sequence of the object of interest is available, as suggested in [GB05]: in their paper temporal information is used on the training phase only to obtain a richer object model, while at run time, only one image is available. Their approach is shown successful on nearly flat objects with highly textured surfaces, with almost no features due to shape. The tracking algorithm they adopt would not allow them to deal with temporary interruptions of the trajectories due to self-occlusions.

Temporal continuity has been explored in a number of application fields. Among them it is worth mentioning action recognition [LL03], video retrieval [SZ03], robotic applications [UGC04]. Our approach fits naturally in the latter application domain, and could be applied to automatic place recognition or robot object grasping.

1.2 Our approach

In this section we briefly describe our approach to the problem of 3D object recognition using temporal information and local descriptors detected in a sequence of images. We represent implicitly 3D information by means of *time-invariant features*, i.e, local features that are distinctive in space and smooth and stable in time. To do so, we extract 2D scale-invariant local features (Harris corners) from the images and track them along the video sequence by Kalman filtering. Kalman filters, robust to temporary occlusions, allow us to deal with general non convex objects, since, while rotating around a 3D object, self-occlusions may cause temporary interruptions in a trajectory.

Then we obtain a compact description of the time-invariant feature by averaging SIFT descriptions for each keypoint belonging to the same trajectory. Each time-invariant feature is made of two distinct parts:

- a spatial appearance descriptor, that is the average of all SIFT vectors of its trajectory
- a temporal descriptor, that contains information on when the feature first appeared in the sequence and on when it was last observed.

Thus we obtain a spatio temporal model made of all these time-invariant features: this model represent the object and its modification due to the motion observed during the sequence. This description of the image sequence content is applied off-line in the training phase on image sequences previously acquired in plain environments, to show the interesting sides of an object.

Once that a model of an object is built, the training phase is finished and for the test phase there are two possible approaches:

1. the off-line approach consists of an analysis of video sequences previously acquired: for each test video, we compute a model similar to those obtained in the training phase
2. the on-line version of this approach consists of a real time analysis: the building of the test model starts when the camera is switched on.

The core of our system is a matching technique [DNOV07] that consists of two steps aiming at exploiting spatial and temporal coherence of time-invariant features: after computing a

first set of matches, the procedure is reinforced by analysing spatial and temporal matches neighborhood.

From the experimental stand point we consider a set of 20 3D objects of different shape and texture complexity. Some of the objects are quite similar to one another. *XXXXX dopo aver scritto la sezione sugli esperimenti aggiustare qui la descrizione degli esperimenti...* We first report results that we obtained on a validation set acquired off-line to validate our spatio-temporal models against scene variations and the growth of the number of models. Then we present results obtained running our real-time recognition system on a variety of settings. The major point of these experiments is to show how, in many different cases, the information obtained from continuous views can greatly help recognition and solve ambiguous matches.

1.3 Structure of the paper

This paper is organized as follows (giusto l'elenco delle sezioni...)

1. Building models with time-invariant features
    - Feature extraction and cleaning
    - The spatio-temporal model
    - The importance of temporal information
    - Space occupancy issue
2. Matching models with spatio temporal models
3. The recognition pipeline
    - Off-line recognition method
    - On-line recognition method
4. Experiments
5. Conclusions

2 Building models with time-invariant features

In this section we introduce the spatio-temporal features around which we base our object recognition method. As we have already mention, one of the main drawbacks of local approaches is that when looking at minute details, an observer may loose the overall object appearance, therefore small details are more likely to be common to many different objects. Another problem that affects local approaches is that when building a model considering many minute details the size of the model can be huge. These observation lead to some of the methods based on geometric, spatial and temporal constraints [FTG06, TMF06, RVG+07] described in Section 1.1.

Let us consider the case in which an object is described by a dense set of views capturing smoothly the appearance variations. Then we can use temporal information to integrate the appearance and spatial description provided by local descriptors. Thus, our aim is to exploit the information hidden in an image sequence to obtain a complete description of the object represented in the sequence itself.

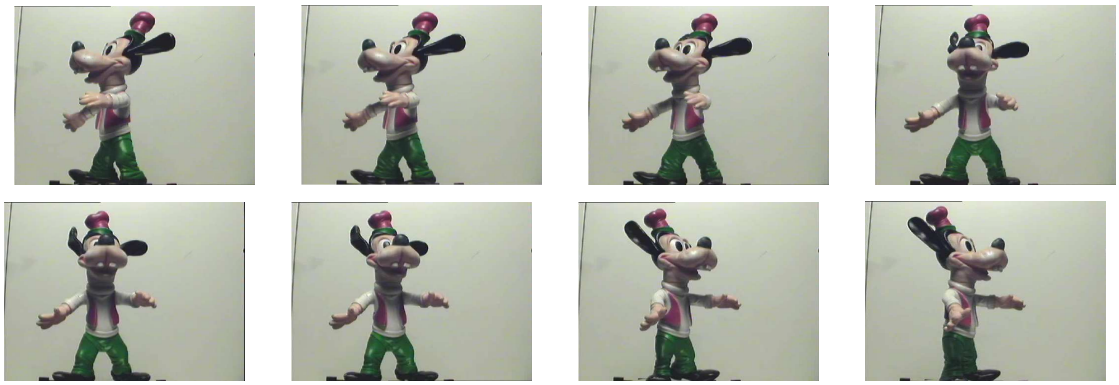

Figure 2: The object *goofy* seen in a sequence capturing its variation in space and time.

If we think about a robot grasping an object and observing it from different points of view it is natural to assume that there are local descriptors which can be spatially and temporally combined. Moreover we can also build a description which is evolving incrementally as the time goes by. Another example in which it is possible to use temporal description, is when the camera moves and observes an object from different points of view. Figure 2 shows an example in which the object *goofy* is seen from different points of view. These are the applications that we have in mind while building our approaches: in such cases we need to model the spatio-temporal

changes of object appearance.

Our method can be briefly described as follows:

- identification of a set of keypoints for each image of the sequence
- description and tracking of the keypoints to capture their variation in time and space
- creation of a model to describe the variation of keypoints appearance
- matching models with spatio-temporal constraints.

Thus, next sections are devoted to explain and give details about the first three steps which are related to the process of building a spatio-temporal model for an image sequence. Finally in Section 3 we will describe the procedure for matching spatio-temporal models of objects.

2.1 Feature extraction and cleaning

We base our recognition approach on the use of local descriptors of images: for each image of the sequence we extract keypoints and describe them using SIFT by Lowe. After the detection and the description of keypoints, each of our descriptors contains the following information about the keypoint k:

$$(\mathbf{p}_k\,,\mathbf{s}_k\,,\mathbf{d}_k\,,\mathbf{H}_k)$$

where $\mathbf{p}_k$ is the position in the space, $\mathbf{s}_k$ refers to the level of scale and $\mathbf{d}_k$ to the principal direction of the keypoint. $\mathbf{H}_k$ contains local orientation histograms around the keypoint. We will use the first three elements for tracking the keypoints while the orientation histograms $\mathbf{H}_k$ will be used as appearance descriptors. It is important to remember that scale and main orientation are also implicitly used for computing the descriptors, as $\mathbf{H}_k$ is built on an image patch centred at the keypoint position, and scaled and rotated according to scale and main orientation.

Since we want to model the variation of keypoints in time and space, we track them and look for a descriptor capable to express the evolution of each keypoint along a sequence. Instead of using a correlation based tracker as the KLT adopted in [Gra04] we choose dynamic filters with prediction capabilities, in particular we start considering the Unscented Kalman Filtering which falls in the class of Particle filters and it is designed for dealing with non linearity of the system.

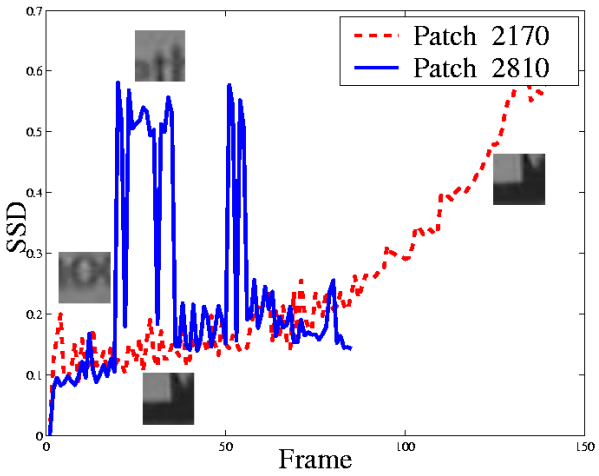

Figure 3: The SSD computed between the patch extracted in the first frame and the following ones. The red line shows the SSD trend for a stable patch while the blue one shows a patch that have an abrupt change due to an error in the tracking.

In our case, the unknown state is defined by the position in space and scale and the principal direction of each keypoint $\mathbf{x}_k = \{\mathbf{p}_k, \mathbf{s}_k, \mathbf{d}_k\}$ For more details on the use of the Unscented Kalman Filtering see [DAOV06] and references therein. Recently we have seen experimentally that, when the sequence is dense enough, a Kalman filtering is faster, accomplishes the need of a stable tracking and it has an easier implementation [DNOV07]. This choice is decisive in the case of the on-line recognition system, when there is the need of higher speed in tracking.

At the end of the tracking procedure, we have many trajectories that can be of variable quality: we may have robust and stable but also noisy and wrong trajectories. Thus there is the need of applying a cleaning procedure that eliminates noisy or wrong trajectories: first, we compute the variance of the scale and of the principal direction of each trajectory. Then, trajectories with a high variance are further analysed in order to check whether they contain abrupt changes that could be caused by tracking errors. To do so, we perform a SSD correlation test between the first gray-level patch of the trajectory and the subsequent ones. In the presence of an abrupt change, the trajectory is discarded. Figure 3 compares two trajectories: the dashed one refers to a good feature, whose appearance varies slowly and smoothly in time; the solid one refers to an unstable trajectory, containing tracking errors (a visual inspection of the gray-level patches confirms this hypothesis).

Figure 4: A visual representation of *goofy* model.

2.2 The spatio-temporal model

The content of an image sequence is redundant both in space and time but we obtain compressed descriptions for the purpose of recognition, extracting a collection of trains of features and discarding all the other information. We call this collection *spatio-temporal model* of the sequence. We do not keep any information on the relative motion between the camera and the object, as it is not informative for the recognition purpose.

More in detail, all the keypoints linked by a tracking trajectory, belong to the same equivalence class. We compute the average of all local orientation histograms $\mathbf{H}_k$, with k running through the trajectory, and we use this as a delegate for all the keypoints of the trajectory. Average values are good representatives of the original keypoints as the tracking procedure is robust and leads to a class of keypoints with a small variance. [Gra04] uses trajectory centroid to select the delegate for each trajectory. This choice is not convenient for our case since it gives the best results with planar object, while our aim is that of representing also keypoints of 3D objects. Thus we decide to use the average as a delegate for the trajectory. Being an average, some histogram peculiarities are smoothed or suppressed, but we will discuss this issue in the next section. It is worth noticing that the descriptors which are too different from the average are discarded to improve the robustness of the descriptor and to eliminate errors due to the tracker.

At the end of this process we have obtained a time-invariant feature that is described by

- a spatial appearance descriptor, that is the average of all SIFT vectors of its trajectory (the *time-invariant feature*);
- a temporal descriptor, that contains information on when the feature first appeared in the sequence and on when it was last observed.

The set of time-invariant features forms the *spatio-temporal model* of the object. A visual representation of the information contained in one model is shown in Figure 4.

We have shown that the tracking phase allows us to obtain a compressed description for the purpose of recognising objects. But there are some cases in which, as we have mentioned above, the *time-invariant feature* tends to oversmooth the description. Thus in the next section we will analyse the effects of changing the length of the train of keypoints.

2.3 The importance of temporal information

In the feature extraction phase, features robust to viewpoint variations are preferred, because we think that a keypoint present in most part of the sequence is more representative than others which are noisy and unstable. Therefore we extract long trajectories. But, if we use the entire long trajectory to compute a descriptor [DAOV06] then we risk to oversmooth the information contained in the trajectory. On this respect it is important to notice that choosing to average a long sequence of descriptors the model is more general, but it can happen that we loose information: since long trajectories are the result of observing a feature on a long period of time (and possibly a high range of views). Thus descriptions generated from these long trajectories tend to oversmooth the appearance information, and may lead to a high number of false positives. This is especially true if the object is deeply 3D and its appearance changes dramatically over the sequence.

To this purpose we apply, before computing the time-invariant feature descriptions, a cutting phase that cuts a trajectory into many sub-trajectories of length N. The choice of N is not crucial, and common sense rules may be applied: if N is too small we loose efficiency and compactness, as the model becomes very big. If N is too big, the information within a feature is oversmoothed. Figure 5 shows the effect of changing N: a small training sequence containing a small range of viewpoints is compared with a test sequence of the same object. A similar

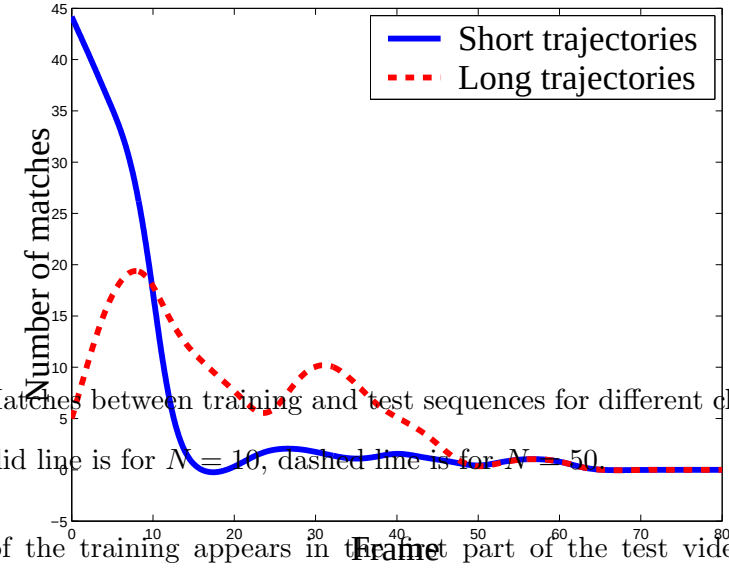

Figure 5: Matches between training and test sequences for different choices of the length of the train N: solid line is for $N = 10$, dashed line is for $N = 50$.

viewpoint of the training appears in the first part of the test video. The solid line shows the matches between training and test models obtained with N=10, the dashed line is for N=50. The smoothing effect as N grows is apparent. To one extreme $N = 1$ is equivalent to discard temporal information and perform image matching, $N = L$ where L is the length of the trajectory is equivalent to avoid cutting.

It is worth noticing that the cutting parameter depends strongly on the quality of the analysed sequence: we noticed that when the sequence of the object is acquired with a better camera as a consequence we obtain better and more stable sequences which allow us to keep a higher value of N. Therefore we can conclude that the choice of this parameter depends also on the quality and the type of the sequences considered. *Infatti durante gli esperimenti che avevamo fatto con la nuova telecamera avevamo visto che poiche' le sequenze erano molto stabili, si potevano ottenere modelli molto piu' compatti non tagliando quasi mai. Ma lo potevamo fare perche' le sequenze erano molto buone... non so se vale la pena dirla sta cosa... boh!?* Then, considering the fact that we devised two kind of recognition approach, it is important to notice that the cutting phase is of utmost importance during the on-line recognition stage. In fact, as we will show in Section 4, to incrementally build a spatio-temporal model, there is the need

to analyze keypoints belonging to a finite temporal window in order to obtain a model which adhere to the scene currently observed. Therefore, the model is updated every 10 frames, which is a different way to cut sequences.

As we have already mentioned in the Introduction, we think that the temporal information is a crucial add for the recognition task. Indeed many biological vision system gather information from motion of head or of the pupil: fixating an unfamiliar object for a few seconds and possibly observing it from slightly different viewpoints is a common practice in human experience. Therefore our approach exploits temporal information to obtain a system that incrementally can model the scene by means of a continuos set of images or a video sequence. We can capture the idea of invariant representation and increase the recognition confidence by observing an object for an amount of time which is proportional to the object peculiarity, its difficulty and the divergence of the acquisition conditions with respect to the training stage.

2.4 Implementation issue

Our approach allows us to obtain a compact representation of the object contained in the sequence. Indeed the representation can be extremely compact in the case we decide to have smoother descriptors or it can contain some more details if we decide to have shorter trajectories for the creation of the descriptors. The advantage of this representation is apparent because the space occupancy of our model is limited. Suppose that the keypoints detected in each frame are approximately 300. Then, if every keypoint of the sequence participates to build the model, in the average case (a sequence of length 250) the model is made of approximately 75000 features. Our approach grants to obtain a compact representation of typically 1500 time-invariant features. A compact model for the object is fundamental when recognising many different objects. *Questo pezzo e' peso... non so come metter giu' il problema della complessita'!* In fact, one of the main problems that we faced is that when the number of models increases, the recognition stage becomes slower, since the matching procedure is performed on a great number of spatio-temporal models. During the off-line recognition, a correct tuning of the parameters and the good performances of the two-stage matching procedure is sufficient to cope with the increase of the models. Since we devised an approach which compare the time-invariant features of the test with those of all the models, the complexity is increasing with N,

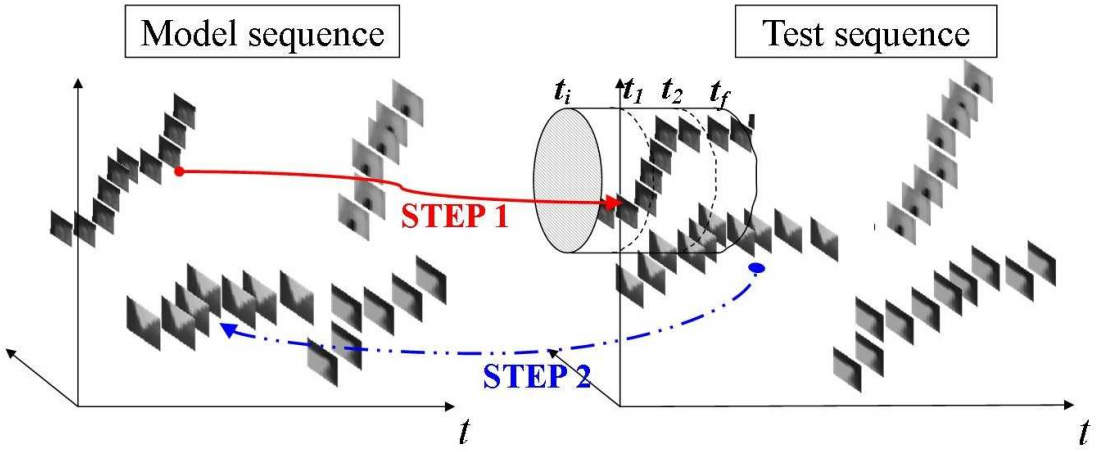

Figure 6: Description of the two-stage matching for comparison of sequences models.

the number of models. A possible solution is to devise a sort of tournament in which one can firstly recognize the type of object (planar or 3D for instance) and then proceed for further more refined comparison. We performed our experiments with a dataset of 20 objects: in the off-line version of our recognition system... in the on-line...

3 Matching models with spatio-temporal constraints

We devise a two-stage matching which exploits spatial and temporal information within each model. It is worth noticing that this matching procedure is performed in the two version of our system both during the off-line and the on-line recognition. Let us consider a test sequence represented by a model T, and a training sequence, represented by an object model M. The problem we address is to see whether the test model T contains the object M.

Our two-steps matching strategy, visually described in Figure 6, is performed as follows: **STEP 1:** We perform a nearest-neighbour matching between each training model and the test model: this procedure gives us initial hypotheses for the presence of known objects in the sequence. For each time-invariant feature in M, we use histogram intersection [SB91] to check whether T contains a similar feature: we set a minimum similarity threshold S_1 to obtain a collection of robust matches $(f_M, f_T)_{S1}$.

STEP 2: This stage, based on enhancing spatio-temporally coherent matches, helps us to confirm or to reject each hypothesis. We use spatio-temporal constraints and perform a backward matching procedure from the test to the training model.

- We detect the subsets of training and test models containing most matches, I_M and I_T, exploiting the temporal descriptors of the features matched after the first step. Doing so, we obtain a raw temporal localization of the object in the test sequence, but also hints about its appearance, since I_M marks a particular range of views. In order to refine this detection, we reject matches which are spatially isolated, founding on their average positions $\overline{p}_k$.

- We increase the number of matches in a spatio-temporal neighborhood around the previously detected features. Let us consider a match $(f_M, f_T)_{S1}$: we can specify two sets, F_M and F_T, containing features (of M and T respectively) appearing together with f_M and f_T for a chosen period of time. A new matching step is then performed between F_T and F_M considering a lower similarity threshold S_2: a pair $(f'_T, f'_M)_{S2}$ is a new match if the features are spatially close to f_T and f_M, respectively.

- This new stage is repeated for each pair $(f_M, f_T)_{S1}$ belonging to I_M and I_T intervals. The incorrect recognition hypotheses are rejected while the correct ones are enriched by further matches in the surrounding areas. In Figure 6 the cylinder bounds a region in which this second search is performed.

This procedure increases the number of high scores when groups of features which are spatio-temporally coherent appear both in training and test. Thanks to the compact representation of visual information the matching phase is efficient even when the sequence is long and there are several detected keypoints.

4 The recognition pipeline

In this section we describe how the matching strategy is applied to recognition, both for off-line and on-line recognition of known objects in the observed scene.

When the recognition system is used off-line the recognition procedure boils down to the matching strategy described in Section 3, applied to each object model and the test model computed from the training and test sequences previously acquired. In this particular case the temporal windows defining the scene of interest are well defined and consist of the whole training and test sequence. We make use of a threshold on the number of matches to decide if a given object is present in one of the test sequences. The recognition process is identical when there is more than one object in the same scene: each model is compared with the test and the keypoints which are correctly matched allow us to identify the corresponding model.

If we move to on-line acquisition our system enters in recognition mode as soon as it starts receiving a video signal. It extracts local features on the first frame and track them on the next frames, incrementally building a model from the video stream. Feature extraction is performed only if the current number of trajectories goes below a given threshold. When, at time t_1, the size of the model reaches a minimal number of spatio-temporal features S_{min}, we look for known objects on the scene: the two-step matching is performed between the test model at time t_1 (that encodes the information from a temporal window $[0, t_1]$) and all training models. Matching can produce one of the following results:

- It exists just one model M (of an object O) producing $W_M \geq W_{min}$ matches while for the other models M_i $W_M - W_{Mi} \geq d$. In this case we reach the conclusion that object O is present in the scene. At this point the system is ready for another recognition, while the temporal window slides ahead.

- More than one model verify the given conditions. We consider such objects as our hypothesis, then we extract new information to improve the description. When the test model is grown of at least N items, we try again the recognition only between the hypothesis;

- There are no models verifying the conditions. The system gives feedbacks to the user suggesting to change scale, in accordance to the training models, or to change observation point. For a fixed delay it continues to increase the model. If no observation is reached it concludes that no known objects are present in the scene after a given number of matching attempts.

For space occupancy and performance reasons the test models need to have a finite memory of

length t: the size of the memory induces a temporal window of analysis that we slide ahead as time goes on. Also, the scene model is reset when a high number of trajectories are abruptly interrupted – this usually happens when the user attention moves from one object to another.

It is worth to notice that when the recognition process is activated real-time on a video stream our system incrementally gathers a first hypothesis on the presence of objects and gives feedbacks to the user on how to explore the world. In the case of multiple matches it encourages the user to observe the object from a slightly different viewpoint. In the case of no matches it suggests to change scale or observation point before concluding that no known object is present in the scene. The system response is incrementally updated and cleaned at run time so that the model of the observed scene does not grow to a size which is difficult to manage.

5 Experiments

5.1 Off-line recognition

5.2 On-line recognition

6 Conclusions

Acknowledgments

References

[BL05] M. Brown and D. G. Lowe. Unsupervised 3d object recognition and reconstruction in unordered datasets. In *Proc. of 3DIM)*, 2005.

[BSP93] D. Beymer, A. Shashua, and T. Poggio. Example based image analysis and synthesis. Technical report, A.I. Memo 1431, MIT, 1993.

[CDFB04] G. Csurka, C.R. Dance, L. Fan, and C. Bray. Visual categorization with bag of keypoints. In *The 8th European Conference on Computer Vision - ECCV*, Prague, 2004.

[CWT00] T. F. Cootes, K. N. Walker, and C. J. Taylor. View-based active appearance models. In *International Conference on Automatic Face and Gesture Recognition*, pages 227–232, 2000.

[DAOV06] E. Delponte, E. Arnaud, F. Odone, and A. Verri. Trains of keypoints for 3d object recognition. In *IEEE International Conference on Pattern Recognition*, Hong Kong, 2006.

[DNOV07] E. Delponte, N. Noceti, F. Odone, and A. Verri. Spatio-temporal constraints for matching view-based descriptions of 3d objects. In *WIAMIS*, 2007.

[ECT99] G. Edwards, T. F. Cootes, and C. J. Taylor. Advances in active appearance models. In *Proceedings of the International Conference on Computer Vision*, 1999.

[EIP] S. Edelman, N. Intrator, and T. Poggio. Complex cells and object recognition. http://kybele.psych.cornell.edu/ edelman/archive.html.

[FTG06] Vittorio Ferrari, Tinne Tuytelaars, and Luc Van Gool. Simultaneous object recognition and segmentation from single or multiple model views. *International Journal of Computer Vision*, 67(2), 2006.

[GB05] M. Grabner and H. Bischof. Object recognition based on local feature trajectories. In *I Cogn. Vision Work.*, 2005.

[Gra04] M. Grabner. Object recognition with local feature trajectories. Master's thesis, Technische Universitat Graz, 2004.

[JP98] M. J. Jones and T. Poggio. Multidimensional morphable models: a framework for representing and matching object classes. *International Journal of Computer Vision*, 2(29):107–131, 1998.

[Lei04] B. Leibe. *Interleaved Object Categorization and Segmentation.* PhD thesis, ETH Zurich, 2004.

[LL03] I. Laptev and T. Lindeberg. Space-time interest points. In *ICCV*, 2003.

[LMS06] B. Leibe, K. Mikolajczyk, and B. Schiele. Efficient clustering and matching for object class recognition. In *British Machine Vision Conference*, 2006.

[Low99] D.G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, pages 1150–1157, Corfú, Greece, 1999.

[Low04] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004.

[MN95] H. Murase and S. Nayar. Visual learning and recognition of 3d objects from appearance. *IJCV*, 14(1), 1995.

[MP05] P. Moreels and P. Perona. Evaluation of features detectors and descriptors based on 3d object. In *ICCV*, 2005.

[MS05] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *Trans. on PAMI*, 27(10), 2005.

[PV92] T. Poggio and T. Vetter. Recognition and structure from one 2d model view: observations on prototypes, object classes and symmetries. Technical report, A.I. Memo 1347, MIT, 1992.

[PV98] M. Pontil and A. Verri. Support Vector Machines for 3D object recognition. *IEEE PAMI*, 20:637–646, 1998.

[RLSP06] Fred Rothganger, Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. 3D object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. *International Journal of Computer Vision*, 66(3), 2006.

[RVG+07] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2007.

[SB91] M. J. Swain and D. H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[SFF07] S. Savarese and L. Fei-Fei. 3d generic object categorization, localization and pose estimation. In *ICCV*, 2007.

[SOP06] T. Serre, Aude Oliva, and Tomaso Poggio. A feedforward architecture accounts for rapid categorization. *PNAS*, 2006.

[SPRP06] S. M. Stringer, G. Perry, E. T. Rolls, and J. H. Proske. Learning invariant object recognition in the visual system with continuous transformations. *Biological cybernetics*, 94:128–142, 2006.

[SZ03] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision*, 2003.

[Tan97] K. Tanaka. Mechanisms of visual object recognition: monkey and human studies. *Current opinion in neurobiology*, 7:523–529, 1997.

[TFL+06] A. Thomas, V. Ferrari, B. Leibe, B. Schiele T. Tuytelaars, and L. Van Gool. Towards multi-view object class detection. In *CVPR*, 2006.

[TMF06] A. Torralba, K. Murphy, and W. Freeman. Sharing visual features for multiclass and multiview object detection. *Trans. on PAMI*, 2006.

[UGC04] A. Ude, C. Gaskett, and G. Cheng. Support vector machines and gabor kernels for object recognition on a humanoid with active foveated vision. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems,*, pages 668–673, 2004.

[VP97] T. Vetter and T. Poggio. Linear object classes and image synthesis from a single example image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):733–742, 1997.