MESH-based Active Monte Carlo Recognition (MESH-AMCR)

Felix v. Hundelshausen and **H.J. Wünsche**
University of the Federal Armed Forces Munich
Department of Aerospace Engineering
Autonomous Systems Technology
85577 Neubiberg, Germany

M. Block, R. Kompass and **R. Rojas**
Free University of Berlin
Department of Computer Science
14195 Berlin, Germany

Abstract

In this paper we extend Active Monte Carlo Recognition (AMCR), a recently proposed framework for object recognition. The approach is based on the analogy between mobile robot localization and object recognition. Up to now AMCR was only shown to work for shape recognition on binary images. In this paper, we significantly extend the approach to work on realistic images of real world objects. We accomplish recognition under similarity transforms and even severe non-rigid and non-affine deformations. We show that our approach works on databases with thousands of objects, that it can better discriminate between objects than state-of-the art approaches and that it has significant conceptual advantages over existing approaches: It allows iterative recognition with simultaneous tracking, iteratively guiding attention to discriminative parts, the inclusion of feedback loops, the simultaneous propagation of multiple hypotheses, multiple object recognition and simultaneous segmentation and recognition. While recognition takes place triangular meshes are constructed that precisely define the correspondence between input and prototype object, even in the case of strong non-rigid deformations.

1 Introduction

Recently, *Active Monte Carlo Recognition* (AMCR), a new object-recognition approach exploiting the analogy between robot localization and object recognition was proposed by [von Hundelshausen and Veloso, 2006]. Although the approach is conceptually interesting, it was only shown to work for shape recognition on binary images, so far. Furthermore the proposed approach is not scalable to large object databases.

In this paper we massively extend the approach in various ways, such that it works on realistic images of real-world objects, that it is scalable to large object databases, and such that a correspondence mesh is constructed between input and prototype object recovering correspondence even in the case of large deformations. We show that our new approach *MESH-AMCR* can better discriminate between objects than state-of-the art methods and that it performs better geometric verification than current approaches for both rigid and deformable objects.

The field of object recognition can be divided in *Object Class Recognition*, the problem of recognizing categories of objects, such as cars, bikes or faces (see [Fei-Fei *et al.*, 2004], [Serre *et al.*, 2005] , [Mutch and Lowe, 2006]) and *Object Identification*, the problem of recognizing specific, identical objects, such a identifying a specific face or a specific building. Our approach *MESH-AMCR* addresses this latter problem.

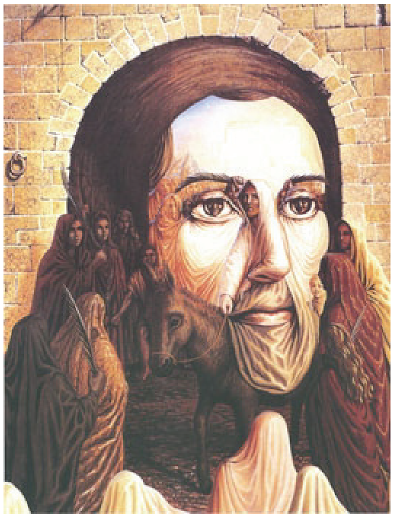


Figure 1: *Palm Sunday*, an ambiguous painting by the Mexican artist Octavio Ocampo

State-of-the-art approaches in object identification typically divide the task in two stages, *Feature Matching* and *Geometric Verification*. Prominent feature detection methods are [Lowe, 2004], [Ke and Sukthankar, 2004], [Mikolajczyk and Schmid, 2004] and [Matas *et al.*, 2002]. For finding prototype images with matching features different schemes have been tried, reaching from nearest neighbor search and Hough-based grouping [Lowe, 2004] to the use of vocabulary trees [Nister and Stewenius, 2006]. In this first stage, the goal is to retrieve a bunch of candidate images, with-out worrying about the geometric validity of feature correspondences. In *Geometric Verification* the geometric layout of feature correspondences is verified. Typically, RANSAC [Fischler and Bolles, 1981] is used to find valid groups of feature correspondences, typically verifying their positions subject to the class of affine transformations or rigid epipolar geometry [Kushal and Ponce, 2006]. An exemption is [Alexander C. Berg, 2005] where low distortion correspondences are found through solving an integer quadratic programming problem, similar to how correspondences where established in the work of [Belongie *et al.*, 2002].

Although, the referenced approaches let us reach a state of maturity in object identification, they still have some significant drawbacks and conceptual problems. With the exemption of [Belongie *et al.*, 2002] their drawback is that they use only the feature descriptors to distinguish between objects. While the features are well-suited to establish hypothetical scale invariant frames of reference, their descriptors only describe a small amount of information that could be used for discrimination. SIFT features, for example, are willingly not detected at edges, but edges at locations where no SIFT-features are detected could help identification. The other drawbacks can best be described by three criterions that should be met by an object recognition framework that is suited to be integrated in a real-time robotic system.

1. **Iterative Recognition** The process of recognition should be distributable over several frames, only performing a small amount of processing in each frame, but this amount being rapidly computable, such that other approaches like tracking can take place simultaneously in real-time. Using the information of several frames over time provides richer information for discrimination. None of the above approaches fulfills this criterion, because they are based on single images and perform all the processing in each image. Only by using an iterative object recognition approach and combining it with tracking the continuity constraints that underly successive images of real-world video can be exploited.

2. **Allowing Feedback** Figure 1 shows, that *the mind sees what it wants*. The nose can either be a nose or an elbow depending on what the mind wants to see. Since recognition is also a question of will" a recognition approach should offer an interface for this will". For example, it should offer an interface of where attention is put on and on being able to specify a bias for favored interpretations. Feedback not only means the control of attention but also on the type of features that are use for discrimination. Also, real motoric actions, like gaze control, or even movements of a robot might be necessary to distinguish two objects. Although these ideas are not new, none of the current best object-identification programs really implements these concepts.

3. **Propagating Multiple Hypotheses** Meanwhile the computer vision community has well understood, that segmentation cannot be accomplished independently from recognition. Thus, no unique ad hoc frame of reference can be established for comparison, but rather different hypothetical matches have to be propagated simultaneously. Rather than performing a depth first examination and iterative breath-first examination has to take place. While different interpretations iteratively compete, this competition can be examined to determine where to guide attention an which features to use in order to discriminate between the competing hypotheses. None of the before-mentioned approaches offers an interface for such a process.

We will show that *MESH-AMCR* allows to integrate all these concepts. The remainder of the paper is organized as follows. In section 2 we shortly review Active Monte Carlo Recognition (AMCR). In section 3 we describe our new approach *MESH-AMCR*. In section 4 we show experimental results. Section 5 finally concludes our paper.

2 Active Monte Carlo Recognition (AMCR)

AMCR was developed as the consequence of recognizing that *mobile robot localization* and *object identification* are essentially the same problem and transferring the best known localization method, Monte Carlo Localization (MCL) [S. Thrun and Dellaert, 2001] to the problem of object recognition[von Hundelshausen and Veloso, 2006]. Similarly to how a robot explores its environment and localizes itself in a set of maps, the focus of attention explores an image and recognition is put as the problem of *localizing the focus of attention* in one of a set of prototype images. In the same way as MCL uses a set of particles to approximate a probability distribution for the robot's position, AMCR uses a set of *M-Particles* to localize the position of the attentive *V-Particle*. Furthermore, not only one point of attention might be used but AMCR employs a set of *V-Particles* to explore the input image, similarly to how an unknown area might be explored by a swarm of robots. To apply MCL only two models have to be provided, a measurement model and a motion model. In AMCR, the measurement model specifies what kind of measurements are taken at the point in the image where a V-Particle is located. The motion model specifies the distribution of how the M-Particles move when the corresponding V-Particle moves. Additionally AMCR requires the specification of a *motion policy*, that is a mechanism of how the V-Particles should explore the input image.

In the original work [von Hundelshausen and Veloso, 2006] only binary images of various shapes where used. Each V- and M-Particle was just described by a (x, y) location together with an orientation. The motion policy was defined by letting the V-Particles move in the direction of their orientation and letting them being reflected at edges. The measurement model was implemented by letting the particles performing a range scan, measuring the distance to the first edge for each scan line, similarly to how laser range scanners are used in mobile robot localization.

This instance of AMCR, *RADIAL-AMCR* is hardly applicable on real word images, since the measurement model is too simplistic. But it has further drawbacks: One is, that the method does not scale well with the number of prototype objects, since for each V-Particle a whole bunch of M-Particles has to be instantiated in each prototype image. Another drawback is that in *RADIAL-AMCR* there was no mechanism to prevent a V-Particle returning to a location it already had explored.

In our new approach *MESH-AMCR* we will get rid of all these drawbacks by specifying a new measurement model, a new motion model and adding the idea of iteratively constructing hypothetical correspondence-meshes.

3 Our new approach: MESH-AMCR

In RADIAL-AMCR a swarm of M-Particles starts forming a cluster in a prototype shape that corresponds to the input image, at a position that corresponds to the respective V-Particle. While the V-Particle is moving in the input image, the M-Particles move accordingly. Thus, a sequence of hypothetically corresponding points in the input image and the prototype images is retrieved. One basic idea of MESH-AMCR is to use these points to construct corresponding meshes. Here, each V-Particle has a triangular mesh, and each of its M-Particles has a corresponding triangular mesh with the same topology, but allowing a deformed geometric layout. Each cell of the meshes is a triangle and each triangle in the V-Particle's mesh has one unique corresponding triangle in each mesh of its connected M-Particles. When MESH-AMCR iteratively converges to a stable interpretation, the mesh of the best M-Particle shows how the input image has to be deformed in order to match the retrieved prototype-image. Here, each triangle in the V-particles mesh is linearly mapped to the corresponding M-Particle's triangle. Indeed the meshes define an affine transformation that can change from cell to cell such that the recovering of non-affine transformations is possible. Before we describe these mechanisms in detail we first solve the problem of the bad scalability to large object databases.

3.1 Candidate Object Retrieval

When using a large database of prototype images, we limit the number of images to be considered by applying the recently proposed vocaulary tree method of [Nister and Stewenius, 2006], but using PCA-SIFT features [Ke and Sukthankar, 2004] instead of maximally stable regions [Matas *et al.*, 2002]. Here, we have to extract all PCA-SIFT features prior to starting MESH-AMCR, and by applying the tree bases scoring scheme we retrieve the k best candidate images. We then apply MESH-AMCR solely on these candidate images. At this point we are violating our own iterative recognition philosophy, but we belief that the PCA-SIFT feature extraction approach can be changed in future, such that the process of extraction can itself be reorganized into an iterative process and distributed over several frames. This would be an interesting topic for future research. Currently, we just extract the features in the first frame, get the candidate prototype images and then start our iterative method with the successive frames. In our experiments we take the best 8candidate images out of a database of 1000 objects.

3.2 V- and M-particles

Both V- and M-Particles will iteratively construct triangular meshes. The trick in MESH-AMCR is that in each iteration a V-Particle corresponds to exactly one cell in the V-Mesh, and each M-Particle corresponds to exactly one cell in the M-Mesh. But the cells that correspond to the particles change in each iteration. In fact, the meshes expand in each iteration by one additional cell, respectively, and always these last cells correspond to the particles. Thus both, V- and M-Particles are basically triangles, defined by three points. This is quite different to [von Hundelshausen and Veloso, 2006] where the particles were just points plus an orientation. In this way each related pair of V- and M-particles naturally defines an affine transformation that represents the hypotheses that the V-triangle has to be linearly mapped to the M-triangle in order to make input and prototype image match to each other. However, the validity of the affine transformation is restricted to the particular cells. Pixels outside the triangles have to be mapped by the affine transformations that are given by the neighboring mesh-cells, if available. Of course, the choice to use triangles as mesh-cells is not arbitrarily: Two corresponding sets of three points constitute the smallest entity that uniquely define an affine transformation.

Summarizing, a V-particle $\mathbf{v}$ is represented as

$$\mathbf{v}_i := (\mathbf{p}_{i0}, \mathbf{p}_{i1}, \mathbf{p}_{i2}, \text{V-mesh}_i), \qquad (1)$$

with $\mathbf{p}_{ik} = (x_{ik}, y_{ik})^T_{k=0,1,2}$, being the corner points of the triangle and V-mesh$_i$ being a triangular mesh that initially only contains a copy of the V-Particle's triangle. Similarly, an M-particle is represented as a triangle, a mesh, and two additional indices:

$$\mathbf{m}_j := (\mathbf{p}'_{j0}, \mathbf{p}'_{j1}, \mathbf{p}'_{j2}, \text{M-mesh}_j, i_j, k_j), \qquad (2)$$

where i_j is the index of the V-particle to which the M-particle is connected and k_j is the prototype image in which the M-particle is located.

3.3 Scale Invariant Features as Igniting Sparks

The M-Particles approximate a probability distribution for the V-Particle's corresponding position and mesh configuration in the prototype maps. Since the meshes are iteratively expanded, the dimensonality of the the space of M-Particles varies. Initially, the meshes start with a single triangle and thus the dimensionality is $6 + 1 = 7$ (6 for the three points and one for the prototype index k_j) for the first iteration. Even this initial dimensionality is very large and a huge number of M-Particles was needed to let recognition converge to the correct solution starting for example from a uniform distribution of M-Particles. To overcome this problem, the trick is to place V- and M-Particles at initial starting position that have a high probability of being correct. We use the PCA-SIFT matches found in the candidate retrieval phase to define good starting points. Here, we place a V-Particle at each keypoint in the input image and connect it to M-Particles placed at possible matches in the prototype images. More precisely, given a keypoint in the input image, we can efficiently navigate down the vocabulary tree reaching a leaf-node that contains a bucket with all keypoints in the protype images that define the same visual word (see [Nister and Stewenius, 2006]). These hypothetical relations between keypoints are the igniting sparks for MESH-AMCR.

A typical initial situtation is illustrated in figure 2. The main notion behind the V- and M-particles is that a pictorial element in the input image can be interpreted in different ways: For instance, the nose in image 2 can either be interpreted as a nose (prototype image face) or as an ellbow (prototype image of an arm). We now define the *motion policy*, the *measurment model* and the *motion model* for MESH-AMCR.

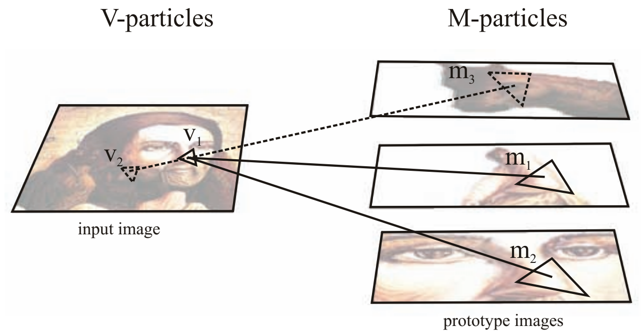


Figure 2: After keypoint matching, V- and M-particles are setup according to probable keypoint matches returned as a byproduct of the candidate object retrieval phase.

3.4 Motion Policy and Mesh Expansion

Consider the case of having only one V-particle $\mathbf{v} := (\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2, \text{V-mesh})$. In each iteration, the V-particle performs a movement, i.e. the coordinates of its three points $(\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2)$ change. This movement is performed in a way, that the new triangle aligns with an edge of the old triangle, thereby expanding one of its edges and successively constructing a triangular mesh. While the V-particle moves in each iteration, the data structure $V - mesh$ holds the corresponding mesh that is constructed. There are many different possibilities regarding the order of edges that are expanded. In our experiments we simply expand the boundary mesh cells in a clockwise order.

With the V-Particle moving and expanding its mesh by one cell each M-particle performs a corresponding movement and mesh expansion. While the topology of the M-mesh corresponds exactly to its V-mesh, the exact position of the points can vary, thereby allowing deformed meshes as is illustrated in figure 3. The distribution of variation is defined in terms of the motion model. Thus while the motion policy defines how a V-particle moves, the motion model describes the variations of movements the M-Particles can perform. Note, that the expansion and position of the triangular cells is independent of the initial keypoints. Only the first cell positions during initialization are placed at the location of keypoint matches.

3.5 Motion Model and Mesh Variations

Because of the resampling process, good M-particles will produce multipe exact copies. When a V-particle performs a movement and mesh expansion step, all the M-particles will also perform a corresponding movement and mesh-expansion step. However, each M-particle will do so in a slightly different way such that variations arise in the newly added triangles of the M-meshes. Indeed, it is only the free point of the new triangle that varies among the M-meshes. However, after several iterations of measurement, resampling and expansion steps, the resulting meshes can be quite different, even when they all started to be the same.

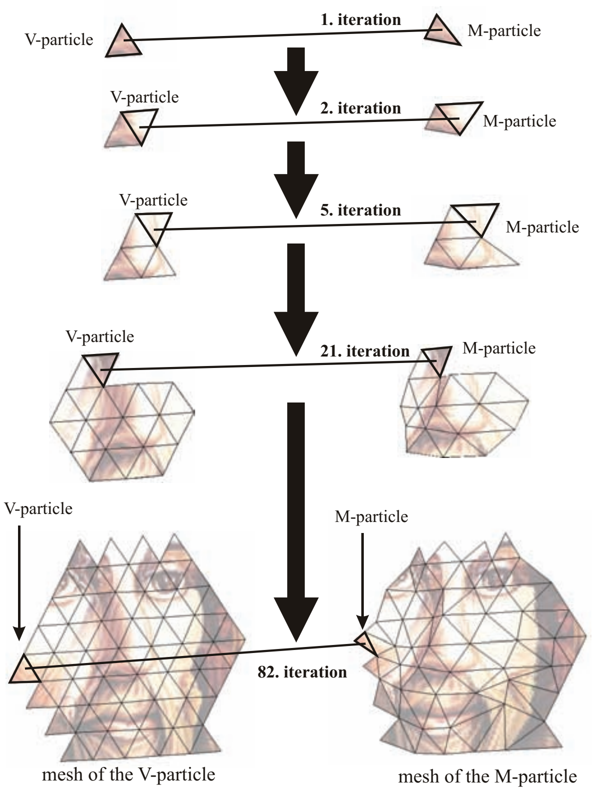


Figure 3: During expansion the meshes behold the same topological structure. However, the positions of the nodes of the M-meshes can vary, thereby allowing to match deformed objects.

3.6 Measurement Model

In each iteration, each M-particle gets a weight that represents the quality of the hypothetical relation to its connected V-particle. This weight is calculated by investigating the appropriate pixels in the input and prototype image. The triangle of the V-particle and the triangle of the M-particle define an affine transformation and we define a dissimilarity function that compares these triangles subject to the given affine transformation. Instead of comparing the pixels, we build histograms with 16 bins of edge orientations and compare these histograms. This idea is inspired by [Edelman *et al.*, 1997], based upon a biological model of complex cells in primary visual cortex.

3.7 Resampling of the M-particles

In each iteration, the M-particles obtain a weight according to the measurement model and the M-particles are resampled. In this process a complete set of new M-particles is generated from the old one. Some of the old M-particles are discarded, and some are copied multiple times to the new set, with the total number of M-particles remaining constant. The expected number of copies of an M-particle is proportional to its weight. Resampling takes place in the set of all M-particles that are connected to the same V-particle, no matter in which prototype they lie. Consequently, it can happen that a prototype looses all its M-particles after some iterations, meaning that the respective prototype does not represent the given input image. Vice versa, M-particles will cluster in prototypes matching the input image. In this way the computational power is concentrated on similar prototypes.

4 Experimental Results

In this section we compare our method with state-of-the-art object recognition systems and show its advantages. Since our goal is object identification we do not compare to object class recognition approaches. We show that MESH-AMCR is better for both geometric verification and distinction between similar objects than existing approaches. To proof this, we show that MESH-AMCR can deal with all cases in which traditional methods work, and that it can additionally solve cases in which state-of-the-art methods fail.

Example

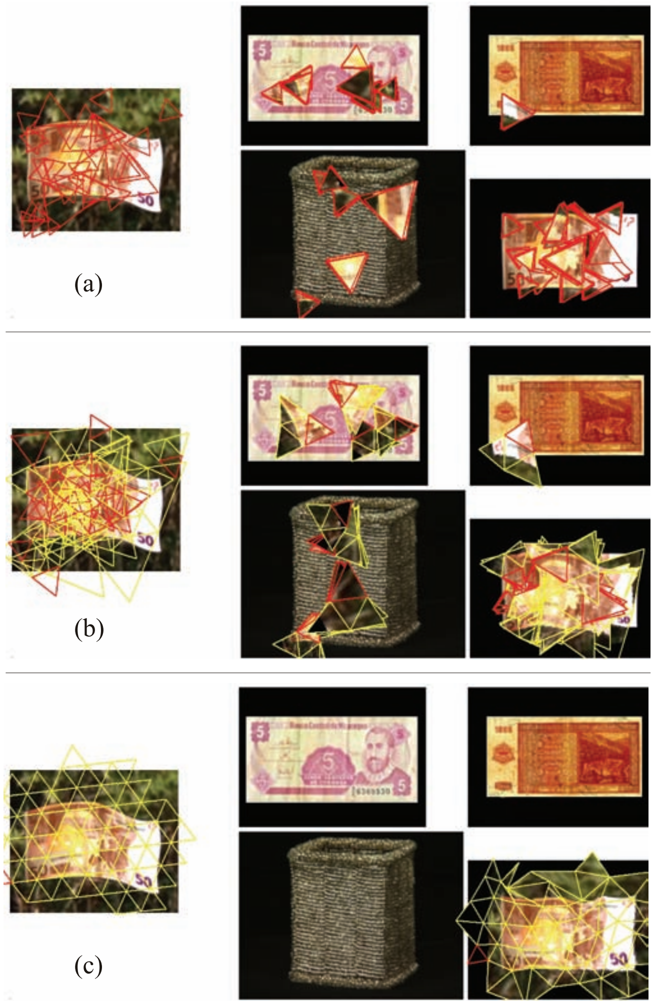
Before describing our results on large object databases we illustrate *MESH-AMCR* with an example. In this example we assume that an image of a deformed banknote has to be recognized within 4 candidate objects retrieved by the vocabulary tree scheme. One of the prototype images contains the same but non-deformed banknote. While we assume that the prototype images are segmented, the input image is not segmented and contains background clutter. Figure 4 shows how recognition is performed and how deformations can be recovered by deforming the meshes of the M-Particles.

Efficient Implementation

Although several hypotheses are propagated, our method is still efficient. The number of V-particles in quite low (10-20). In each iteration only one triangular cell of each M-Mesh has to be compared to its corresponding cell of its V-Mesh. Since several M-Particles are linked to the same V-Particle, the pixels of each V-Particle's cell have to be sampled only once. Comparing the triangular cells subject to their affine transformation can be efficiently done using standard graphics accelerator hardware (e.g. using DirectX).

Experiments on rigid object databases

To evaluate our method we use the wide-baseline stereo dataset of the Amsterdam Library of Object Images (ALOI) [Geusebroek *et al.*, 2005] containing realistic images of 1000 objects, each captured from three different viewpoints (left, center, right) rotated by 15 degrees each. We used all 1000 right images to learn the vocabulary tree. As test images we used the left images (differing by 30 degrees) but additionally we scaled them down to 40 percent of the original size, plus rotating each about 45 degrees. For the geometric verification phase, we implemented a method of reference according to current approaches like [Lowe, 2004] and [Ke *et al.*, 2004]. They use affine transformations to explain the geometric layout for keypoint matches. In our reference method, we use RANSAC [Fischler and Bolles, 1981] to verify the geometric validity of keypoint matches based on affine transformations. Using this approach we get a recognition rate of 72.8 percent on the 1000 objects database. For comparison we applied MESH-AMCR on the same dataset and test images. Here we used only 6 iterations leading to hypothetical correspondence-meshes with 6 triangular cells, respectively. As recognized object we take the candidate-image that has the most M-Particles at the end. In doing so, we get a recognition rate of 85.2 percent. Thus, MESH-AMCR is performing significantly better. MESH-AMCR correctly recognized 46.1 of the cases that were falsely classified by the affine transforma-


(a)

(b)

(c)

Figure 4: Red (dark) triangles show the position of V- and M-Particles. The yellow (light) triangles show the corresponding meshes. The pixels within the M-Meshes are texture mapped from its corresponding V-Meshes. In (a) V- and M-Particles are initialized according to well matching PCA-SIFT features. (b) shows the situation after 3 iterations. In (c) only the maximum a posteriori (MAP) estimate of the M-Particle and its corresponding V-Particle is shown. The meshes of the MAP V- and M-particles specify the correspondence between input and correctly identified prototype image.

tion based RANSAC method.

These failures typically occur when the image of one of the objects - input object or prototype object - has only few keypoints. In the work of [Ke *et al.*, 2004] 5 keypoint matches are required as minimal configuration that satisfies geometric constraints. In the approach of [Lowe, 2004] as few as 3 keypoint-matches constitute the minimal valid constellation. This design was made, to allow recognition of objects with few keypoints. Unfortunately, the less keypoint-matches the more likely the chance of finding a matching group in a wrong prototype image.

The advantage of our method is that geometric verification is not based on the keypoint descriptors but on the original image data referenced by the constructed correspondence-meshes. Thus, our source of information is much richer for both discrimination and geometric verification. We only need one keypoint-match as igniting spark for mesh-construction at the beginning. Thus, while requiring less keypoint-matches to initiate verification, we still have richer information to check for geometric constraints.

5 Conclusions

In this paper, we have proposed *MESH-AMCR*, a new algorithm for object identification that has its origins in methods for mobile robot localization. Besides the advantages of better discrimination and recognition of deformable objects our approach has some conceptual advantages: First, recognition is performed in an iterative way such that other algorithms like tracking can run in parallel. Second, it does not assume a prior segmentation of the input images. Third, it is able to propagate several hypotheses simultaneously. Hence, the simultaneous recognition of several objects is possible, too. Additionally, the algorithm yields precise correspondence in form of corresponding mesh-cells. There are plenty of possibilities of extending the approach: One of the most interesting one is how to guide the V-Particles to parts that let us best discriminate among competing hypotheses, and how to dynamically adjust the measurement models of the V-Particles for best discrimination. Another interesting question is of how the meshes could be used to recover shading models to allow larger variations in lighting conditions.

References

[Alexander C. Berg, 2005] Jitendra Malik Alexander C. Berg, Tamara L. Berg. Shape matching and object recognition using low distortion correspondence. In *IEEE Computer Vision and Pattern Recognition (CVPR) 2005*, 2005.

[Belongie *et al.*, 2002] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522, 2002.

[Edelman *et al.*, 1997] S. Edelman, N. Intrator, and T. Poggio. Complex cells and object recognition. unpublished manuscript: http://kybele.psych.cornell.edu/edelman/archive.html, 1997.

[Fei-Fei *et al.*, 2004] Li Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. pages 178–178, 2004.

[Fischler and Bolles, 1981] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[Geusebroek *et al.*, 2005] J. M. Geusebroek, G. J. Burghouts, and A. W. M. Smeulders. The Amsterdam library of object images. *Int. J. Comput. Vision*, 61(1):103–112, 2005.

[Ke and Sukthankar, 2004] Yan Ke and Rahul Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2004)*, volume 2, pages 506–513, 2004.

[Ke *et al.*, 2004] Y. Ke, R. Sukthankar, and L. Huston. Efficient near-duplicate and sub-image retrieval. In *Proceedings of ACM Mulimedia*, 2004.

[Kushal and Ponce, 2006] Akash Kushal and Jean Ponce. Modling 3d objects from stereo views and recognizing them in photographs. In *European Conference on Computer Vision (ECCV)*, 2006.

[Lowe, 2004] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[Matas *et al.*, 2002] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *BMVC*, pages 384–393, 2002.

[Mikolajczyk and Schmid, 2004] Krystian Mikolajczyk and Cordelia Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.

[Mutch and Lowe, 2006] Jim Mutch and David G. Lowe. Multiclass object recognition with spars, localized features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, page to appear, 2006.

[Nister and Stewenius, 2006] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR 2006*, 2006.

[S. Thrun and Dellaert, 2001] W. Burgard S. Thrun, D. Fox and F. Dellaert. Robust Monte Carlo Localization for Mobile Robots. *Artificial Intelligence*, 2001.

[Serre *et al.*, 2005] Thomas Serre, Lior Wolf, and Tomaso Poggio. Object recognition with features inspired by visual cortex. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005)*, pages 994–1000, 2005.

[von Hundelshausen and Veloso, 2006] Felix von Hundelshausen and Manuela Veloso. Active monte carlo recognition. In *Proceedings of the 29th Annual German Conference on Artificial Intelligence. Springer Lecture Notes in Artificial Intelligence*, 2006.