*Computer Vision Winter Workshop 2008, Janez Perš (ed.)*
*Moravske Toplice, Slovenia, February 4–6*
*Slovenian Pattern Recognition Society, Ljubljana, Slovenia*

# Robust Online Object Learning and Recognition by MSER Tracking

Hayko Riemenschneider, Michael Donoser, and Horst Bischof

Institute for Computer Graphics and Vision,
Graz University of Technology, Austria
(hayko, donoser, bischof)@icg.tugraz.at

**Abstract**

*This work presents a robust online learning and recognition system. The basic idea is to exploit information from tracking an object during the recognition and/or learning stage to obtain increased robustness and better recognition results. Object tracking by means of an extended MSER tracker is utilized to detect local features and construct their trajectories. Compact object representations are formed by summarizing the trajectories to corresponding frontal MSERs. All steps are performed online including the MSER detection, tracking, summarization, SIFT description as well as learning and recognition based on a vocabulary tree. The proposed method is evaluated on realistic video sequences which prove the increased performance for robust online recognition. The whole system runs at a frame rate of 9 fps on a standard PC.*

## 1 Introduction

In the past robust learning and recognition required some form of offline processing to cope with the large amount of required training data for the complex learning algorithms [12]. In this paper we propose a robust system which handles learning and recognition online providing state-of-the-art recognition rates. Most object recognition systems ignore the fact that usually a short sequence of the object is available. In our system training is handled by a tracking algorithm which pursuits and learns the visible sides of an object. In [13] it is suggested that a connection between continuous views improves the recognition capabilities. The association between various appearances of the same object is realized by tracking. Tracking features on the object provides a better learning experience by carefully learning the best views and summarizing these to an online retrievable object representation.

Maximally Stable Extremal Regions (MSER) [6] are used as interest regions. Matas et al. proposed this detector for wide-baseline stereo matching and defined extremal regions which possess properties such as affine transformation invariance, multi-scale detection, and a fast enumeration. Evaluations [8] show that MSERs are detecting stable features of arbitrary shape and scale. Despite the number of detected regions being low, the repeatability is better than with other detectors especially for viewpoint changes [7] and their efficiency is unbeaten [8].

The learning of various views involves tracking an object by means of MSER tracking [1] which delivers an efficient and accurate matching of MSERs between consecutive frames. The tracking is used to construct trajectories to comprise all of the collected motion information and appearances of an object. These trajectories are then evaluated on their tracking quality and summarized to robust and compact object representations. Previous work [3, 4] used the relative change of a SIFT descriptor to detect a stable minimum where the trajectory contains the best representation. In this work we introduce the notion of *frontal MSERs* for optimal representation.

These object representations are described by a SIFT descriptor [5] and stored for later retrieval. For this purpose a vocabulary tree data structure [9] is incorporated to efficiently insert new online learned objects and also to recognize objects during the tracking.

The proposed system uses tracking for learning as well as for recognition to compact the appearance of an object. The combination of state-of-the-art detection, tracking, robust summarization, description and retrieval techniques provides the necessary boost to perform all steps in an online process. Figure 1 illustrates the proposed system.

The outline of the paper is as follows: In Section 2 we describe the object tracking algorithm used to build trajectories. Section 3 shows how the object representation is learned by summarizing the trajectories. The recognition process is described in Section 4. Experimental results for recognizing learned objects are presented in Section 4 and finally, conclusions are drawn in Section 5.

## 2 Object Tracking

The idea behind tracking for object recognition is to follow the motion of an object to learn all sides and select the best representations for each feature independent of the view at which it is detected. The tracking process incorporates the notion of Maximally Stable Extremal Regions (MSER) Tracking [1], which enables efficient and accurate tracking by using information from previous frames to reduce the search space in consecutive frames. The tracking step is the process of finding a new extremal region which best fits the previously tracked MSER. A vector consisting of its size, center of mass, stability and intensity range is used to determine the best match which is then used as next MSER in subsequent images.

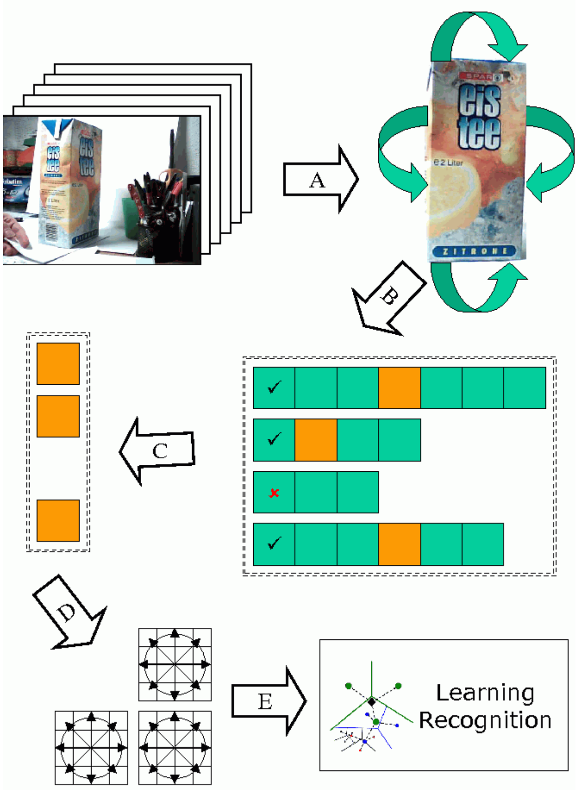


**Figure 1:** The system consists of five steps: (A) Tracking, (B) Extraction of trajectories, (C) Frontal MSER selection, (D) SIFT description and (E) either learning or recognition by means of a vocabulary tree.

The MSER tracking notion as described by [1] ensures robust tracking of all MSERs identified in the initial image by considering all similar extremal regions of an image as tracked representations. Additionally, there are three speedups which are proposed to allow faster tracking. First, only one type of detection – either `MSER-` or `MSER+` – is performed. Second, only an adapted part of the analysis is performed through limiting the gray value range of the image. Each MSER which represents an extremal region is attributed two specific gray values defined by the brightest and darkest pixel included in its region. And finally, the search area within the image for matching possible extremal regions is reduced significantly by looking in a location near its predecessor.

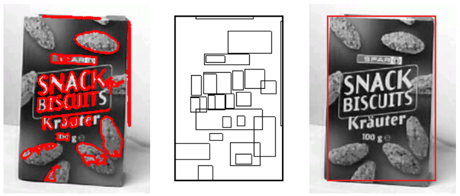
The benefit of tracking is the accurate matching between consecutive frames which again is used to construct trajectories. A trajectory is a set of features tracked over time. The wealth of information due to the tracking is later used to build a compact object representation.

## 2.1 Compound MSER Tracking

Since a single MSER is not sufficient for good recognition, multiple MSERs are considered using an extension denoted as *compound MSER tracking*. This method provides significant advantages over the previously known *single MSER tracking* and *color MSER tracking* [2]. These methods fail on highly inhomogeneous objects since these are not always robustly detected. The resulting match is considered unstable and results in an incorrect segmentation and unwanted analysis of the background.

The novel *compound MSER tracking* is suitable for tracking multiple MSERs simultaneously and is illustrated in Figure 2. This method detects and matches multiple smaller MSERs directly. It is an extension of the *single MSER tracking* to a compound analysis. Each tracked MSER is only analyzed in an image region around its previous center of mass and a range in gray value intensities both tightly restricted by its predecessor. This provides the full benefits since it speeds up the detection, increases the accuracy of the matching process, and third, only analyzes the image regions of interest.

The bounding box of the individually tracked MSERs is combined to a global bounding box which is then used for segmentation and restriction for the further robust tracking. Further robustness is achieved through evaluation of the behavior in motion and properties of the tracked MSERs.



(a) Input (b) Compound (c) Global

**Figure 2:** In *compound MSER tracking* individually detected MSERs (a) and their bounding boxes are combined a global bounding box (b) which is used as a restriction for next tracking steps and redetections (c).

## 2.2 Stable Matching and Redetection

First, a first-order motion model is used to derive a specific direction of motion. The image region is restricted to a region of interest (ROI) of the previous stable match given by its bounding box. This ROI is extended in only the direction of the detected motion. This provides a closer search region inside the image space than a non-directional extension. As consequence less neighboring MSERs are incorrectly matched and confused with one another.

Second, after the best available match is determined through comparing the vector of MSER properties by an Euclidean distance, the stability of the new MSER is evaluated. A threshold is applied to verify that its stability value – the relative change in size – is sufficiently small for stable tracking. If the minimally required stability is not reached, the previous MSER is reused instead of the new unstable one. However, if only unstable or no best matches at all could be found for a number of times, the MSER is dropped from tracking and its trajectory ends. The measure used is a robustness counter which is increased each time the stability is insufficient. It is decreased when the matching result is stable enough. This mechanism allows for small repairs during tracking. Figure 3 gives an overview of the length of trajectories and the effects of repairing. To minimize the effect of various motions and their frame rates it shows the frequencies of track length on average for several sequences. When

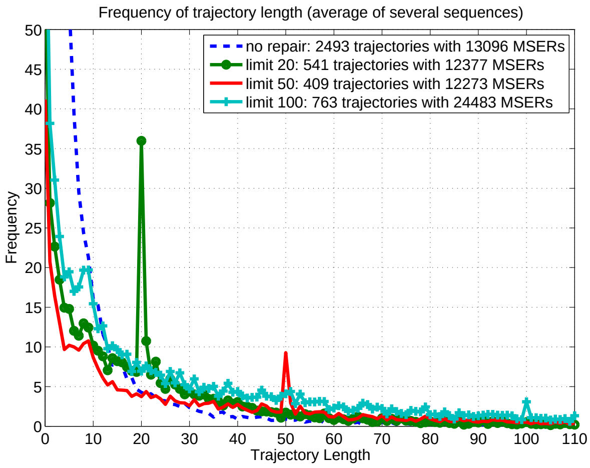


**Figure 3:** Comparison of the track length frequencies: The robustness limit is set to various values showing its effect on repairing trajectories by accepting unstable matches which would otherwise terminate trajectories immediately (no repair). The robustness limit evaluates the changes in tracking stability. Trajectories are longer when this limit is set to a higher value by repairing unstable periods during tracking. Please note, for better comparison the maximum frequency shown is 50 whereas no repair results in initial frequencies up to 1000 but quickly drops to 50, as shown here.

no robustness limit is applied there is a large number of trajectories each with very short track lengths. This means no repair is performed and the tracking fails to robustly match MSERs over longer periods of time. A selection of increasing limits in Figure 3 shows the respective increase in track lengths.

A high robustness limit repairs more trajectories but also allows tracked MSERs to remain instable for a long time. Due to a limited number of concurrently tracked MSER, a prolonged tracking of unstable MSERs prevents the detection of new MSERs. This deteriorates the information collected through the trajectory and the learning quality.

The third step is a further evaluation on the behavior of the trajectories. The aim is to quickly terminate trajectories when the tracked MSER has suddenly become very unstable while still maintaining a good stability value, i.e. a small change in relative size. This step thus includes a set of rules evaluated frame by frame. These include maximum limits on absolute size, relative size increase, a change in location, as well as a check for duplicates.

Due to the frequent termination due to lost stable matches the number of active trajectories decreases steadily. A full frame redetection is costly in terms of computation time and does not take into account that the object has been tracked so far. Therefore, we use the bounding box of the currently stable MSERs to provide a ROI for new detections. Once the total number of tracked MSERs is low, a redetection is performed only on the reduced image space. The threshold of active trajectories is set to a balance between retrieving new features frequently and an acceptable processing time.

Additionally, a novel solution to merging currently active trajectories with new detections is proposed. It is efficiently solved by ignoring the active trajectories during the detection. The process involves removing the shapes of the current MSERs by ignoring their pixels during the analysis for new MSERs. Figure 4 illustrates this process and its steps. First, the image is cropped to the bounding box of interest. Second, the previous MSERs as shown in b) are subtracted from the image resulting in a reduced image such as c). Third, the reduced area is analyzed for MSERs. The combination of any new detection d) is guaranteed to be non-overlapping e) with the previous MSERs.

Thus a time-consuming comparison and merge algorithm was replaced by a further reduction of search space. This speedup also provides less duplicate detections and ensures that other previously uncovered regions of an object are also analyzed for new features frequently.

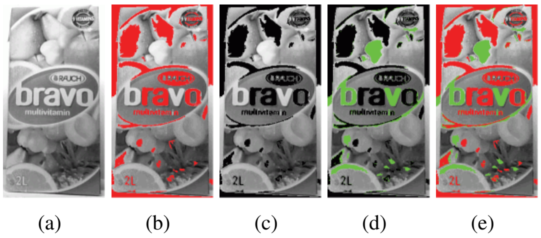

(a) (b) (c) (d) (e)

**Figure 4:** Illustration of new detection: (a) Input image, the previously tracked MSERs (b) are removed from the input to a reduced image (c), newly detected MSERs (d) are thus guaranteed non-overlapping and efficiently merge (e) with the previously tracked MSERs.

## 3 Object Representation

Tracking provides trajectories which describe the motion and appearance of each MSER on the object. This information is used to build a robust and compact object representation. The online learning and recognition requires a representation to be repeatable and compact.

The information acquired through tracking is enormous and redundant. Every trajectory contains every MSER tracked through the length of its active trajectory. Refer to Figure 3, the number of trajectories is about 540 and the total number of tracked MSERs is roughly 12400. The goal of the object representation is to reduce this wealth of information and provide a robust and compact subset.

First, the trajectories are evaluated to select a set of robust trajectories. Second, each robust trajectory is summarized to a single representative denoted as *frontal MSER*. The SIFT descriptor [5] is used to describe this final subset of *frontal MSERs*. The resulting descriptors make up the robust and compact object representation which sufficiently distinctive to apply it for online learning and recognition.

### 3.1 Robust Trajectories

Tracking provides an ongoing evaluation of the trajectories and ensures that only stable trajectories are pursuit. Due to online performance tracking only contains a few features simultaneously. If no evaluation is present, the tracker is quickly trapped with instable trajectories and is not able to track new MSERs. Thus it is a requirement of the tracker itself to perform an evaluation which is also used as robust selection for object representation.

The final robust subset is selected based on the quality of the trajectory. The quality is measured by

$$quality = \frac{stable\ matches}{tracking\ length} \qquad (1)$$

where the number of successful and accurate matches is divided though the tracking length for normalization. A threshold for this quality value is set and a second threshold for the minimum track length is used to determine a robust subset.

### 3.2 Compact Representation

The next step involves finding a suitable summarization for a trajectory. In a single image system there exists only individual features without trajectories. The benefit of tracking and building trajectories is to collect more information about the features detected on an object. The information is then used to find a meaningful compact representation of the entire trajectory.

Previous work [3, 4] used the relative change of a descriptor to detect the best representation within a trajectory. For a controlled rotation the minimum can be detected by a quadratic fit ignoring outliers, whereas for an arbitrary rotation or generally any uncontrolled movement the minimum is harder to detect.

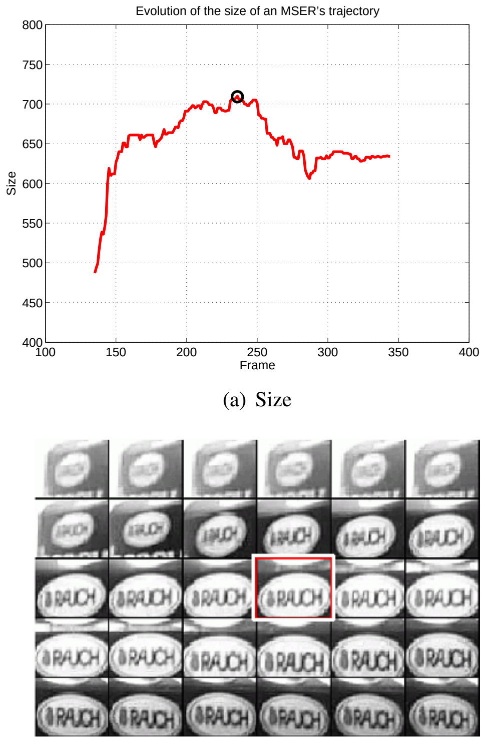
Since we are using MSERs as feature detector more information than just position and orientation of a SIFT descriptor is available. An MSER has an arbitrary shape which reflects the perspective distortion in which it is viewed. The most suitable view for a compact representation is described by [3] as the one which is parallel to the viewing plane. The feature – in our case the MSER – does undergo perspective distortion and if viewed at an angle the distortion decreases the size of the MSER. This property is used to select the *frontal MSER*, i. e. the MSER providing the frontal view.

Figure 5a shows a graph of one exemplary trajectory and its evolution in size over track length. For this trajectory the maximum is clearly detectable, as indicated by the circle. This selection is used to identify the *frontal MSER*. In Figure 5b the evolution of a subset of a trajectory is shown. This illustration supports the choice of the MSER with the maximum size as suitable representation for its trajectory. The increase in size reduces not only the perspective distortions but also improves the quality of the underlying image data. As is seen the tracking delivers a larger more clearly recognizable *Rauch* logo which is used as summarization for this trajectory.

To complete the object representation the selected *frontal MSERs* are normalized [10, 11] to achieve affine invariance, which has less importance now that the least perspectively distorted view is chosen. Finally SIFT descriptors [5] provide repeatable and distinctive vectors of feature descriptions.

## 4 Object Recognition

At this stage an object and its trajectory have been tracked and a distinctive object representation exists. The SIFT descriptors are used to learn and identify the object. The recognition requires that the descriptors are compared to all other object representations and provides a distinguishing vote for the desired object.



(a) Size

(b) Logo

**Figure 5:** This figure shows in (a) the evolution of size of an MSER over the course of its trajectory. In (b) a subset of the about 200 frames long trajectory is shown.

For this purpose the vocabulary tree by Nistér and Stewénius [9] is used to store and retrieve the descriptions and object information. This method is based on a tree data structure which borrows ideas from text retrieval systems. The two main benefits of the hierarchical structure are the minimal computation requirements for inserting new objects and matching unknown objects against the entire vocabulary tree. Second, the number of objects stored in the data structure does not affect the recognition time significantly. Thus the same approach is ready to be extended to a much larger learning and recognition system with possibly thousands of objects.

The vocabulary tree is an efficient representation of the clustering of SIFT descriptors. The approach uses k-means clustering for each level of the tree. This achieves a hierarchy of clusters which again is used to efficiently traverse the vocabulary tree and find matching cluster centers, referred to as nodes.

Each node contains an inverted file list which is an index to the objects whose descriptors are included in this node. Further each node contains a weight based on entropy. The more objects are included in a node the less distinctive it becomes. Nistér and Stewénius define various voting strategies for retrieval. A *flat strategy* considers only the leaf level for scoring. In the *hierarchical strategies* the scoring is based on how many levels upwards from the leaf level are also considered during scoring. The *flat strategy* is fast while the second improves the recognition rate significantly.

The final score is determined by the sum over all weights of nodes where the query descriptors matches. The frequency of matches for each descriptor is used and normalized by the total number of descriptors – for the query object and the already known objects in the vocabulary tree.

### 4.1 Online Insertion

The hierarchical design of the vocabulary tree allows for a fast insertion of new objects. For each of its descriptors the top nodes and their cluster centers are matched. Only the children of the best matched cluster center are then matched again. This reduction of search space allows for a complete search of the vocabulary in $k \times l$ comparisons. During the insertion of a new object this advantage is used to find the best matching leaf node quickly. For each of the descriptors such a match is sought. Then, a new object identifier is included into the nodes' inverted file list and its weight is updated. No further steps are required.

For the learning step the object is tracked and its trajectories are recorded. *Frontal MSER* selection and the evaluation of the tracking quality during the tracking provide robust and compact trajectories. The affine normalization and SIFT descriptors are created and then inserted into the vocabulary tree as a new object.

### 4.2 Online Retrieval

The same hierarchical matching is used to determine the best matching nodes for retrieval. Due to the lower computational expense only flat scoring is used and no levels other than the leaf nodes are considered for scoring. The result of the retrieval is a list of objects which matched the query object in respect to the nodes the objects share. If a query object matches a node, all objects in its inverted file are possible retrieval matches and are considered. The list of objects contains the final score over all matched nodes using all descriptors.

In the recognition task the object is equally tracked and robust trajectories are summarized to *frontal MSERs*. However, multiple recognition steps are performed at certain intervals. Depending on the required response rate these intervals may range from once per frame to every 20th frame. The recognition matches their SIFT descriptors against the learned objects contained in the vocabulary tree.

### 4.3 Confidence Measurement

The final step of the recognition system evaluates the score retrieved through the vocabulary tree. The score provides a measure how many nodes and descriptors are successfully matched for each considered object. While a simple maximum score selection is the straightforward approach, it is not useful in this case.

Due to the tracking the more information is learned the longer an object is tracked. Initially only a few features are tracked and described. Thus the retrieved score is based on a small number of features. The two benefits of tracking now show their effect in score. First, the longer the tracking the better the selected *frontal MSER*. Second, the longer the tracking the more features are visible and detected.

The idea is to create a confidence measure which evaluates how stable and accurate the recognition of the top score is. The proposed measure is defined as

$$confidence = \frac{highest\ score}{second\ highest\ score}. \tag{2}$$

This distance ratio determines how similar the top two scores are. If there is enough distance between these the recognition is likely to be correct. To determine such a threshold the experiments includes a setup where this confidence is evaluated. In this case the highest score is taken to be correct score.

## 5 Evaluation

In this section experimental evaluations demonstrate the capabilities of our system for recognizing objects learned through trajectories. Each of the test objects has been tracked, summarized, described and inserted into a vocabulary tree clustered and filled with a subset of the UK Bench image database [9] as described later on. The evaluation framework is part of the complete online learning and recognition system and is briefly described. The experiments consist of three sub-experiments. First, the recognition performance of the trajectory tracking is compared to a single view full frame test scene and a ROI of the test scene. Second, the increase of the recognition score during the tracking is evaluated to derive a threshold for the confidence measure. Finally, the confidence decision is evaluated on the test scenes.

The robust learning and recognition system is used in an offline fashion to ensure repeatability. The processes remain the same except for the image source which is provided through pre-recorded video files. For each task a video of an unknown object is presented together with an initialization ROI around the object. During the course of the experiment the same videos with the identical bounding boxes are used for each type of experiment.

For the learning step the object shown by the video is tracked and summarized. In the recognition task the entire video is equally analyzed, however at certain intervals the recognition step is performed. This confidence measure determines the certainty of the recognition result.

### 5.1 Training and Testing Data

The set of videos used for learning and recognition show five different objects. Figure 6 shows the first frames of several sequences for the five objects.

Each of the objects has a range of unique visual features, however at the same time the objects share similar aspects such as letters, symbols and shapes. All objects contain text on their surfaces and some objects share the same letters in a similar font. Figure 6 also illustrates the range of viewing and lighting conditions. In each sequence the object undergoes motion in an arbitrary way including a combination of rotation around the y-axis and in-plane, translation and shearing. Scaling is intentionally avoided to ensure the concept of proper *frontal MSER* selection. Thus the motion is performed at a similar distance from the camera position.

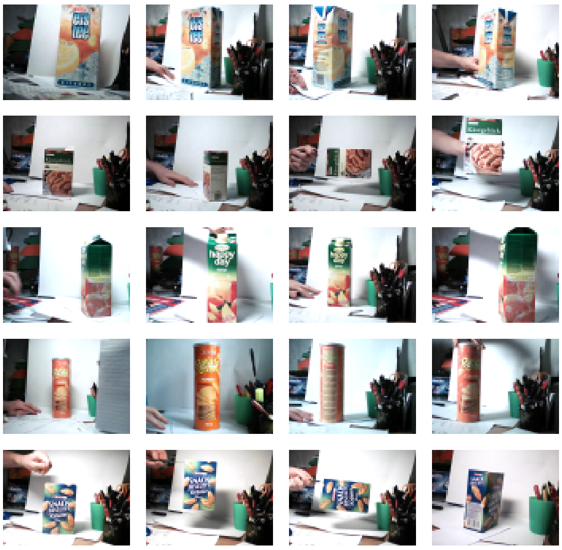


**Figure 6:** This is an overview of the five different objects used for the experiments and a subset of the 34 videos used for evaluation. Each time the first frame is shown indicating the various viewing conditions.

Due to the focus on evaluating the combination of tracking and trajectories only few objects are investigated. A data structure such as vocabulary tree however is designed for a larger number of objects. To approximate more realistic test conditions the vocabulary tree is filled with random objects from the UK Bench [9] database. In total 100 images are used which corresponds to 25 objects at four different viewpoints each. This does not provide an optimal setup. However, the focus is set on evaluation of the trajectories and not the size of the database. The vocabulary tree structure is build for nine clusters and four levels. Using the flat scoring strategy an average top score of 3.54 is achieved which is similar to results by Nistér and Stewénius [9].

## 5.2 Experiment 1 - Recognition Methods

This experiment evaluates the recognition performance comparing the results obtained by tracking objects to single frame recognition. Each of the test objects has been tracked, summarized, described and inserted into the vocabulary tree.

The recognition rate is evaluated in five setups where the only difference is the method of feature extraction. First, the same tracking and trajectory summarization is used to extract the features. Second, a full frame where the object is shown in its dominant frontal position is analyzed for features. Third, the same frame is cropped to a ROI around the object of interest. Fourth, a non-frontal view of the object is used for a full frame feature extraction. And fifth, again the ROI around the object in this non-frontal view is used.

The comparison of full frame to a cropped ROI has the main intend to provide an equal basis to the tracking approach, since both of these are initialized with a bounding box roughly separating the object from its background.

Table 1 shows the results of this experiment. The recognition rate is determined by the relative number of correctly identified objects as best match. The columns from left to right represent the results for tracking through an entire video of roughly 300 frames and the recognition rate after the 100th frames). The next two columns show the performance for the frontal or non-frontal views provided by single images. Full frame and bounding box results are combined in this table since the recognition rate is identical for both.

| Objects | Tracking | | Frontal | Non-Frontal |
|---|---|---|---|---|
| | Full | 100th | Scene | Scene |
| Eistee | 100% | 83% | 100% | 25% |
| Geback | 100% | 100% | 50% | 0% |
| Happyday | 86% | 83% | 75% | 0% |
| Pringles | 100% | 100% | 0% | 50% |
| Snack | 100% | 57% | 100% | 100% |
| Total | 97% | 83% | 69% | 42% |

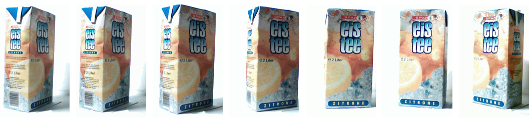
**Table 1:** A comparison of tracking against single frame recognition. The percent of correct recognition is shown for the full video, after the 100th frame (object motion starts after 30 frames), and for a frame of a frontal view and a non-frontal view of the object.

The advantage of the object tracking is clearly shown. The analysis of frontal views shows that enough information is available to match 69% of the test scenes. Since learning only uses the *frontal MSERs* to create an object representation, the frontal view of an object provides a clear view of these MSERs without tracking. However, features on other sides are hidden which explain the lower recognition rate than for the tracking. Similar holds true for the non-frontal views. Here the perspective distortion is too large to be accurately modelled by affine transformation to provide similar visual representations and descriptions as learned by tracking. Less than half of the scenes are detected correctly.

## 5.3 Experiment 2 - Recognition over Time

In this experiment the progress of the recognition score is investigated in terms of time. The expected result is that the longer an object is learned, the better its recognition score will be.

The goal of a robust online learning and recognition system is the ability to cope with arbitrary motion of the object. To reflect this situation the video sequences have been recorded in a similar fashion. The following sections provide an overview of two main motions and an analysis of their recognition score over time.



(a) 0 (b) 50 (c) 100 (d) 150 (e) 200 (f) 250 (g) 350

**Figure 7:** Pure Frontal Rotation Motion: Selected frames of the video sequence showing the rotation. Around frame 200 the object is presented at its dominant frontal view.

### 5.3.1 Pure Frontal Rotation Motion

In the sequence illustrated in Figure 7 there exists only a rotation of the object from a non-frontal to a frontal and again to a non-frontal view. The evolution of the recognition score is shown in Figure 8 a). The general course of the score is as expected and the final score is more than twice that of the second best score.

There are several interesting parts in Figure 8 a). First, the spike at frame 100 and the subsequent drop to a much lower score. This is explained through Figure 8 b) which shows the evolution of the size of the tracked MSERs. For illustration purpose only 25 trajectories with a minimum length of 150 are drawn, which makes up roughly 15% of the selected robust and 3% of all trajectories. At frame 70 two trajectories arrive at their maximum size and thus frontal position, as indicated by the circle. Additionally, two more trajectories commence their tracking. At frame 100 these two new MSERs arrive at a relatively stable size. That means, at least four new frontal MSERs are available for recognition. This boosts the score to the new peak score.

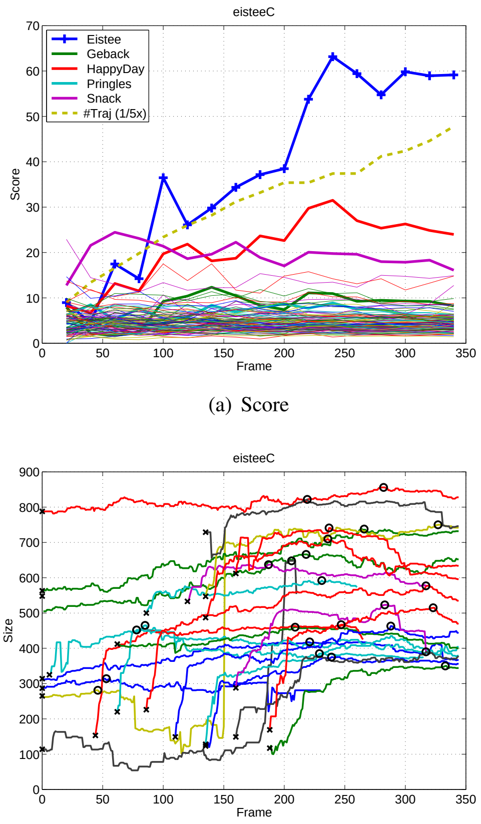
The slight drop afterwards is due to new trajectories which do not resemble the best frontal view but which are taken into account when normalizing. Another aspect may be the case when a good frontal MSER is incorrectly matched and the new frontal MSER is no longer part of the representation of the learned object. In Figure 8 a) this effect is seen at frame 120 when the correct score drops and the score of another object increases suddenly.

The second interesting part of Figure 8 a) is the distribution of scores of the unrelated objects. The thick lines which are also shown in the legend are the new objects learned through trajectory summarization. The thin lines indicate the scores for UK Bench images. When the number of robust trajectories is still low, as illustrated by the dashed line, the few matches which occur during the vocabulary tree matching process have a much greater influence of the score. This explains why another object has a higher score than the correct object up to about 100 robust trajectories. This is also a common situation in other video sequences where the correct recognition also receives the highest score after at least 100 robust trajectories.

The third effect which is visible in Figure 8 a) is the clustering of the UK Bench images at the lower spectrum of the score while four out of the top ten scores belong to the newly learned objects. The content of the two image types varies greatly and thus provides an advantage.

**5.3.2 Towards Frontal View Rotation** In this experiment the sequence as shown in Figure 9 starts off with a non-frontal view and after half of the view the dominant frontal view is shown. From the on only a slight translation is performed.

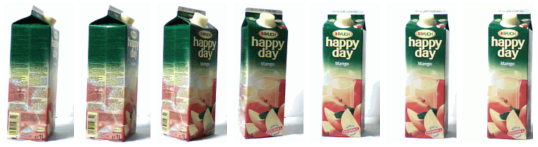
The evolution of the score as illustrated in Figure 10 reflects this motion. The initial view does not provide good MSERs for recognizing this object. The score behaves similarly to the preloaded UK Bench objects while other learned objects receive a slightly higher score. Starting with the detection of MSERs from the frontal view the score steadily increases up to a stable level.



(a) Score

(b) Size

**Figure 8:** Pure Frontal Rotation Motion: a) The recognition score and b) the size of the tracked MSER both in relation to the frames.



(a) 0 (b) 50 (c) 100 (d) 150 (e) 200 (f) 250 (g) 300

**Figure 9:** Towards Frontal View Rotation: Selected frames of the video sequence showing the initial view and the rotation towards the dominant frontal view around frame 150.

## 5.4 Experiment 3 - Confidence

In this third experiment the influence of the tracking length on the recognition is analyzed. This is done by evaluating the confidence measure introduced in Section 4.3 for all video sequences. Figure 11 shows a graph of the confidence along with the minimum confidence value for correct recognition as indicated by the thick dotted line. The analyzed confidence value indicates how much higher the correct score is with respect to the second highest score. If the confidence falls below the dotted line, the recognition is incorrect because another object has a higher score.

Except for five sequences the confidence is already above the minimum for a correct recognition. The final of the five sequence achieves the confidence level after about 150 frames. It is the same as used in Section 5.3.2 where the initial score is very low. Thus for the online use a minimum confidence of 1.2 and minimum score of ten is applied.

Many other confidence measures are available and it would be valuable and interesting to derive a second measure evaluating how many of the robust trajectories have

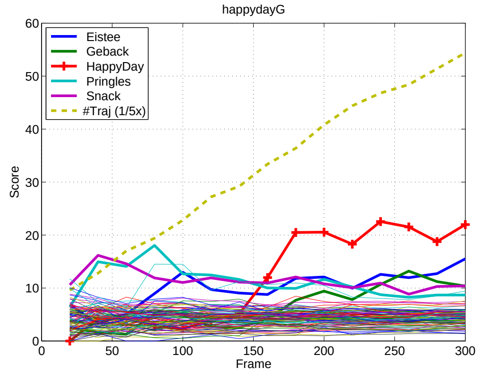


**Figure 10:** Towards Frontal View Rotation: The recognition score with the dominant frontal view around frame 180.

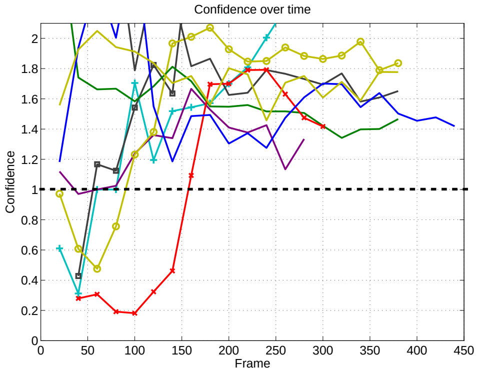


**Figure 11:** This graph shows the confidence measure applied to the video sequences, for better viewing only a representative subset is drawn. The thick dotted lines indicates the minimum confidence for a correct detection.

reached their globally optimal *frontal MSER*. However, other measures are not required as this method provides the necessary decision power to create a robust online learning and recognition system.

## 6 Conclusion

This paper proposed an online robust learning and recognition system which uses tracking to improve the recognition performance. The approach is able to perform all tasks related to learning and recognizing objects in an online manner. The significant gain in performance was demonstrated in the experiments. The online processing is possible due to the novel concepts of *compound MSER tracking*, robust trajectory selection and most importantly the efficient and optimal summarization into *frontal MSERs*.

The introduced system can be improved by combining both types of MSER detection into the tracking process. Also the notion of additional representatives [14] may be used for better representation of large changes in the evolution of a trajectory's appearance. The next steps however will be a combination of multiple MSER tracking methods to provide a better shape segmentation or construction of 3D models.

## References

[1] M. Donoser and H. Bischof. Efficient Maximally Stable Extremal Region (MSER) Tracking. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 553–560, 2006.

[2] M. Donoser, H. Bischof, and M. Wiltsche. Color Blob Segmentation by MSER Analysis. In *Proceedings of International Conference on Image Processing (ICIP)*, pages 757–760, October 2006.

[3] M. Grabner. Object Recognition with local feature trajectories. Master's thesis, Technical University in Graz, October 2004.

[4] M. Grabner and H. Bischof. Object Recognition based on local feature trajectories. In *Proceedings of the International Cognitive Vision Workshop (ICVW)*, 2005.

[5] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. In *International Journal of Computer Vision (IJCV)*, volume 60, pages 91–110, November 2004.

[6] J. Matas, O. Chum, U. Martin, and T. Pajdla. Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. In *Proceedings of British Machine Vision Conference (BMVC)*, volume 1, pages 384–393, 2002.

[7] K. Mikolajczyk and C. Schmid. Scale & Affine Invariant Interest Point Detectors. *International Journal of Computer Vision (IJCV)*, 60(1):63–86, October 2004.

[8] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A Comparison of Affine Region Detectors. In *International Journal of Computer Vision (IJCV)*, volume 65, pages 43–72, 2005.

[9] D. Nistér and H. Stewénius. Scalable Recognition with a Vocabulary Tree. In *Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 2161–2168, June 2006.

[10] S. Obdržálek and J. Matas. Local Affine Frames for Image Retrieval. In *Proceedings of the International Conference on Image and Video Retrieval (CIVR)*, pages 318–327, 2002.

[11] S. Obdržálek and J. Matas. Object recognition using local affine frames on distinguished regions. In *Proceedings of British Machine Vision Conference (BMVC)*, volume 1, pages 113–122, 2002.

[12] P. Roth, M. Donoser, and H. Bischof. Tracking for Learning an Object Representation from Unlabeled Data. In *Proceedings of the Computer Vision Winter Workshop (CVWW)*, pages 46–51, 2006.

[13] G. Wallis and H. Bülthoff. Effects of temporal association on recognition memory. In *Proceedings of the National Academy of Sciences (PNAS)*, volume 98, pages 4800–4804, April 2001.

[14] C. Wallraven and H. Bülthoff. Acquiring Robust Representations for Recognition from Image Sequences. In *Proceedings of the DAGM-Symposium on Pattern Recognition*, pages 216–222, 2001.