OnEye – Producing and broadcasting generalized interactive videos

Alain Pagani[1], Christian Bailer[2], Didier Stricker[3]

[1,2,3] German Research Center for Artificial Intelligence DFKI GmbH, Kaiserslautern, Germany

E-mail: [1]alain.pagani@dfki.de, [2]christian.bailer@dfki.de, [3]didier.stricker@dfki.de

Abstract: **Interactive videos where objects are enriched with additional information have several important applications including e-commerce, education and gaming. However, the production of such videos is difficult and costly due to the lack of tools to automatize the necessary tasks. In addition broadcasting such videos still remains an issue as current video players do not incorporate the possibility to add supplementary media content. In this paper, we present OnEye, a framework that allows video producers to make objects clickable in their videos and to easily incorporate additional content to the video. The framework consists of different tools that support the creation of such enriched media along the production chain up to the visualization by the end-user. The technologies involve state of the art tracking methods and intelligent user interface, as well as web-based player capabilities. We present an application scenario based on online shopping.**

Keywords: Interactive videos, clickable videos, embedded advertising.

1 INTRODUCTION

Digital video is the driving force behind the expansion of the webTV, IPTV and the new "Generation Mobile". To cope with actual trends, non-promotional and affordable collections of digital videos should be made attractive to the user. The new way for advertising arises rather from the placement of products within a scene, leading to a replacement of the classical TV commercials. This activity is already used in television and cinema and is known as "product placement". Thus, the watching experience is not interrupted, and still the products are presented to the audience. The next development step is called "embedded advertising": an object within a scene “contains” the advertisement, and the viewer can select the object in order to view additional information (such as product photos, specifications, etc.), and may also order the article. This combination of embedded advertising and e-commerce is referred to as t-commerce. But the successful application of this business model presupposes a good use of the technology of object tracking within the digital video to track over longer sequences and to allow selection by the viewer.

In this paper, we introduce tools to create such enriched video content and to present them to the audience in a specific way. We show that a vision based object tracking can help in the generation process, but also that an interactive process is necessary. Furthermore, we present a new video player capable of reading enriched video content over the web. Our technology called “OnEye” is composed of three elements (see Figure 1): OnEye Creator is a web-based software that allows for editing standard videos, tracking objects and creating hyperlinks for tracked objects. The outputs of OnEye Creator are enriched videos that we call OnEye Videos. They contain encrypted supplementary information in form of an XML file. These videos can be read by a specific player called OnEye Player, which is available for Desktop, Tablets and Smartphones.

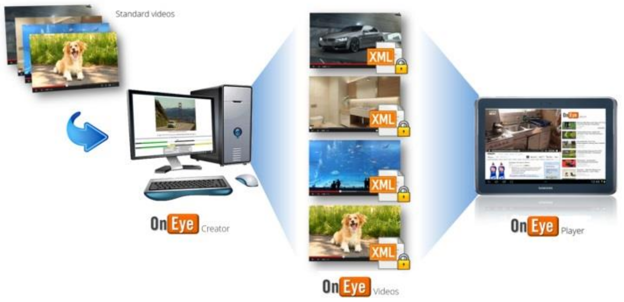

Figure 1: The three components of the OnEye system: OnEye Creator, OnEye Videos and OnEye Player.

The remainder of the paper is organized as follows: In Section 2, we review existing systems and discuss the requirements of a production tool for interactive videos. We then present the software OnEye Creator in the light of the provided tracking methods and the intelligent user interface in Section 3. Section 4 presents the Videos and the Player. We present the results of our evaluation in Section 5 before concluding and addressing future work.

2 RELATED WORK

In the recent years a lot of effort was put into creating full automatic object tracking approaches. An overview can be found in overview works like [1, 2] or tracking evaluation works like [3, 4]. By contrast, only very few works addressed the problem of semi-automatic tracking, although full automatic tracking is still not reliable enough for many practical applications. One semi-automatic framework is presented by Bertolino et al. [5]. Their tracking algorithm is segmentation based, and it can create very accurate results with exact object borders, as long as it tracks correctly. The user’s task is to initialize the segmentation and to correct it if it gets erroneous over time. To fulfil the task the application provides the user

Corresponding author: Alain Pagani, German Research Center for Artificial Intelligence, Trippstadter Straße 122, 67663 Kaiserslautern, Germany, +49 631 205 75 3530, alain.pagani@dfki.de

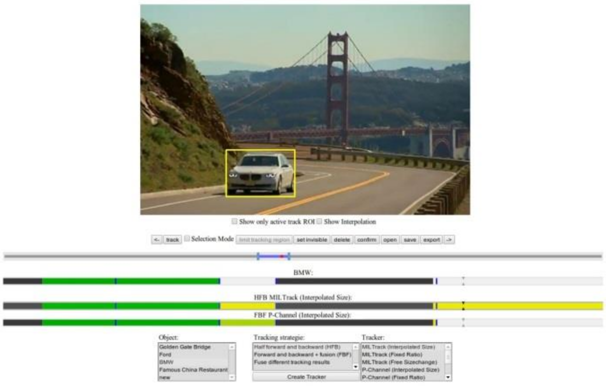
□ Show only active track ROI □ Show Interpolation
<-
track
□ Selection Mode
limit tracking region
set invisible
delete
confirm
open
save
export
->
BMW:
HFB MILTrack (Interpolated Size):
FBF P-Channel (Interpolated Size):
Object:
Golden Gate Bridge
Ford
BMW
Famous China Restaurant
new
Tracking strategie:
Half forward and backward (HFB)
Forward and backward + fusion (FBF)
Fuse different tracking results
Create Tracker
Tracker:
MILTrack (Interpolated Size)
MILTtrack (Fixed Ratio)
MILTtrack (Free Sizechange)
P-Channel (Interpolated Size)
P-Channel (Fixed Ratio)

Figure 2: The Graphical User Interface

several frame based editing tools. Further semi-automatic segmentation based tracking approaches that work similar can be found in [6] and [7]. For these approaches however, a complete segmentation of the object has to be provided, which is often not necessary for clickable videos, and needs a time-consuming human supervision. In our approach in contrast, we optimize the time consumption by providing intelligent tools in order to minimize the user interaction while guaranteeing verifiable correct results. Actually, the requirements for producing reliable clickable videos are quite different from the requirements of classical object tracking. In the object tracking literature, researchers implicitly aim at solving the full-automatic tracking problem, which can be defined as follows: given one single view of an object of interest (usually in the first frame of a video sequence), follow this object throughout the sequence despite possible appearance changes. Results of such papers usually show that in some cases it is possible to follow an object for a time frame ranging from a few seconds to a few minutes. Benchmarks [4] have shown that no tracker is able to track reasonably every sequence, and the best trackers can count on a success rate not exceeding 80% (and less, depending on the sequence). In the production of clickable videos in contrast, the requirements are reversed: the result must be 100% correct, whatever it costs. This means that at least a last verification by a human user is indispensable before validating the results. In our approach, we make use of this human intervention, but recognizing that human resource is costly, we explicitly aim at minimizing user interaction while guaranteeing perfect results. Existing applications like Clikthrough [8] or VideoClix [9] use manual intervention and sell video processing as a service (pay-per-video). WireWax [10] is the only currently available video edition software that allows for interactive object tagging. In this software, faces are automatically detected and tracked, but the tracking of other objects is difficult due to the lack of user intervention and validation. Our system aims at bridging the gap between automatic tracking and full manual tracking by providing the right tools to the user.

3 ONEYE CREATOR

3.1 Object multiple selection and tracking

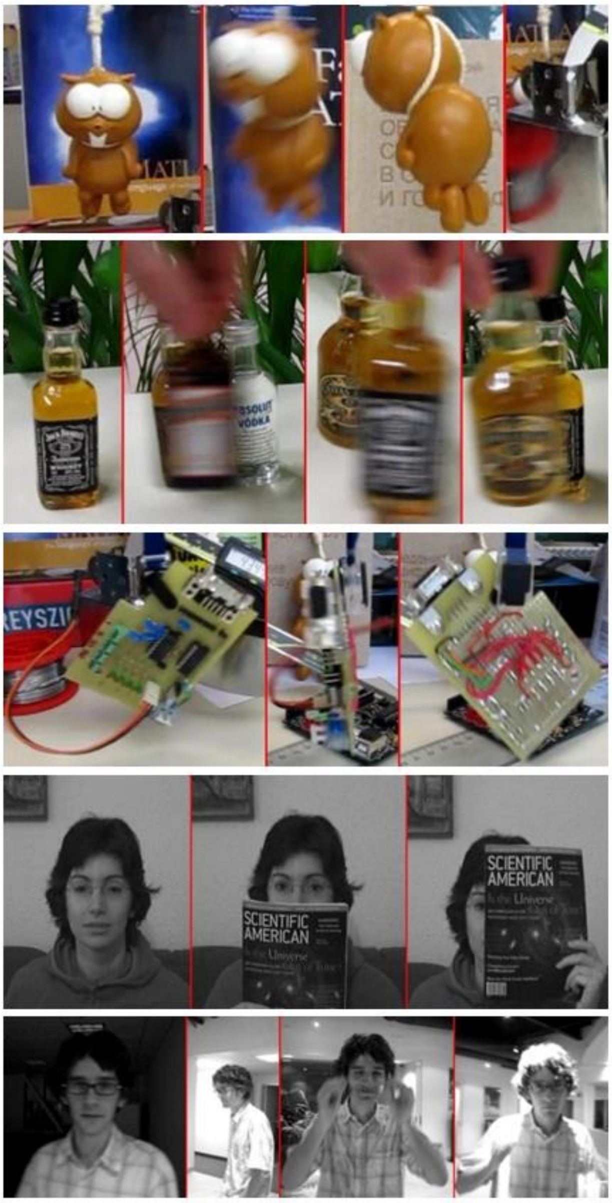

Figure 3: The sequences used in the evaluation. From top to bottom: “Lemming”, “Liquor”, “Board”, “Faceocc” and “David”

In this section we describe our video edition software for object tracking, OnEye Creator. The software consists in a server-side application and a web-based client. The client implements the graphical user interface, and the tracking algorithms are running on a distant server. Figure 2 shows the graphical user interface for video edition: In the upper half, the video is shown as in a standard player. A timeline allows the user to seek a given frame or to play /fast forward or rewind the video. The user then has the possibility to provide examples of the considered object in one or several so-called reference frames by simply drawing a bonding box around the object of interest. We call the frames where the object has been selected by the user **user-specified frames** or **USF**. Note that the user will usually provide several USFs for a given object over the full sequence. This contrasts with classical automatic tracking, where the user usually provides only the position of the first frame of the sequence. This allows us to develop advanced tracking strategies that exploit the multiple user input.

Once a sufficient number of USFs have been entered by the user, automatic tracking can take place. We have implemented 6 of the best state-of-the art object trackers in the system in a modulable way, so that more and more trackers can be added to the system. The available trackers so fare are the following: General methods. The general tracking methods we implemented are an object detection-based methods that builds upon the idea of the P-Channel representation [11], a modified version of the MILTrack algorithm [12] (with HAAR and HOG features), Visual Tracking Decomposition (VTD) [13] and Circulant Structure with Kernels (CSK) [14]. Additionally, we implemented two specialized methods: the first one is a color based tracker that is extremely reliable if background and foreground can be separated, and the second other one is a blob tracker that works only with static background.

3.2 Necessity of a multi-tracker approach

We conducted a study with standard sequences in order to characterize the different trackers and to evaluate the feasibility of tracker selection. The results of the evaluation are detailed in Section 5. The outcome of the evaluation showed, that no existing tracking algorithm was able to track successfully an object automatically in all the sequences. Some trackers seem to be specialized for specific cases, and some sequences are too complex to allow for automatic tracking. However, in our application, we are seeking guaranteed 100% correct results. We therefore implemented all trackers in the software and allow the user to try many trackers at the same time for object tracking. It is convenient to use different trackers on the same sequence, and we have developed an intelligent tracker fusion mechanism based on an adapted majority voting that can automatically select the best tracker among the tested ones to ensure better tracking results.

3.3 Exploiting multiple user input

Because the user can select the object to track several times over the complete sequence, we have an advantage over standard automatic tracking methods. We currently exploit this advantage as follows: we first split the sequence into subsequences starting at a USF and ending at a USF. For each subsequence, we can apply one of these strategies: (1) Track forward from the starting USF until the middle of the sequence and backward from the ending USF until the middle of the sequence. This proves to add robustness when compared with standard (single-direction) tracking. (2) Track forward until the end of the subsequence and backward from the end to the beginning and compare the outputs of each direction. We then alert the user with a color-coded timeline whenever the two directions tracks differ, and he/she can add more USF in the differing parts. (3) With multiple USFs, we can interpolate the trajectory of the object between USFs using a 2D B-spline. Here again, we use a color coded timeline for indicating compliance (green) between the interpolation and the track or differences (red). The user can then rapidly go to the timeframe where interpolation and track differ in order to add one or more USFs in the critical timeslots. Thus, the user can iteratively converge to the correct track. Once the user is satisfied with the results of a tracker, he/she can *validate* a single frame or a range of frames. These validated frames are called **user-validated frames** or **UVF.** Here again, a color code on the timeline of the video easily show which parts of the sequence have been successfully processed, and which ones remain to track and validate. The goal is to validate all the frames of a video sequence.

3.4 Exporting tracking results

The procedure of interactive tracking can be repeated for as many objects as wanted in a given video sequence. For that, the user simply chooses “new object” in the menu of the software and can start to define the track of the second object. Once all objects of the sequence have been tracked, the results can be exported to an XML file that stores for each frame and each object the position of the bounding box of the object. After exporting, the video file is enriched with extra information about object location that can be used for making clickable video.

4 ONEYE VIDEOS AND PLAYER

An OnEye Video is a video file that contains the location of one or several objects over the sequence as supplementary information. In the current version of our software, this information is encoded in a separate XML file, but in future versions we will encode it directly into the video file. In order to use this information, OnEye Videos can be played in a specific Player – the OnEye Player. Our player is implemented in HTML5 and Javascript and can play the video in the same manner as the standard players, while providing the extra possibility to interact with the selected objects by clicking on the object. Once a click has been detected on a pre-defined object, an event is launched – usually the video pauses and an object-specific information is shown on or besides the video. In our prototype, this is implemented by adding an URL to each object in the accompanying XML file. When an object is clicked, the URL of the object is opened besides the video in a mini-browser. This solution is generic and allows for different kinds of content being loaded by clicking.

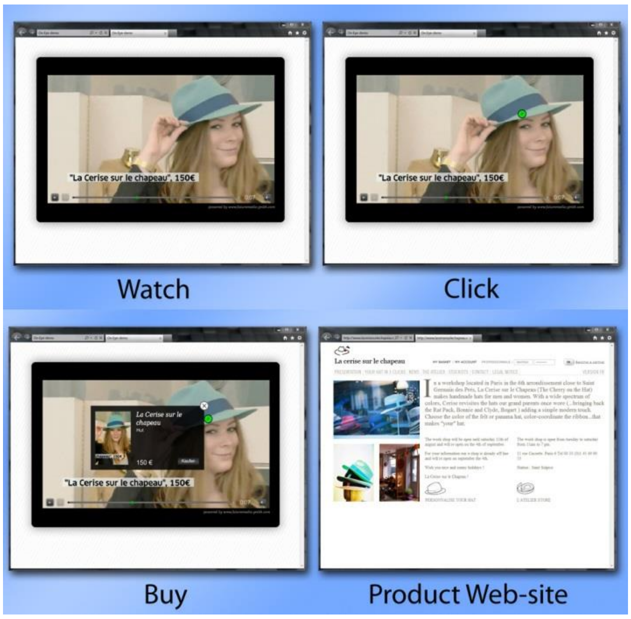

Figure 5: View of the OnEye Player in a commercial scenario

5 EVALUATION OF EXISTING TRACKERS

The first idea of the project was to select the best possible tracker for generic object tracking in video sequences. We therefore implemented and evaluated different state-of-the-art tracking methods and compared their output on different representative sequences.

For this experiment, we took 5 sequences usually used for tracker evaluation: "Lemming", "Liquor", "Board" from the PROST Dataset [15], "Faceocc" from the FragTrack Dataset [16] and "David" from the IVT Dataset [17] (see Figure 3 for exemplary frames from these sequences). For each sequence, we tracked the object of interest with all the following methods: MILTrack with HAAR features (HAAR), MILTrack with HOG features (HOG), MILTrack with both HAAR and HOG features (HAAR+HOG), MILTrack with Color HOG features (CHOG), MILTrack with HOG features without online learning (only the first frame is taken into account in the appearance model)(HOGffo), P-Channel and VTD. The result of tracking is shown in compliance diagrams: for each frame we measure the compliance between the bounding box found by the tracker B_{track} and the ground truth GT by computing the overlap as follows:

$$o = \frac{B_{track} \cap GT}{B_{track} \cup GT}$$

The diagrams in Figure 4 show in the x-axis a threshold of the overlap o and on the y-axis the percentage of frames of the sequence that attain at least an overlap of value o. If we take an overlap threshold of 0.5 or 0.6, we see in these experiments that the tracking algorithm that works best for a specific sequence usually performs poor on other sequences, and that no single tracker produces acceptable tracking results for all the sequences. It was therefore necessary to adopt a strategy where many different trackers are used, with an intelligent fusion of tracker results as well as a user-initiated validation.

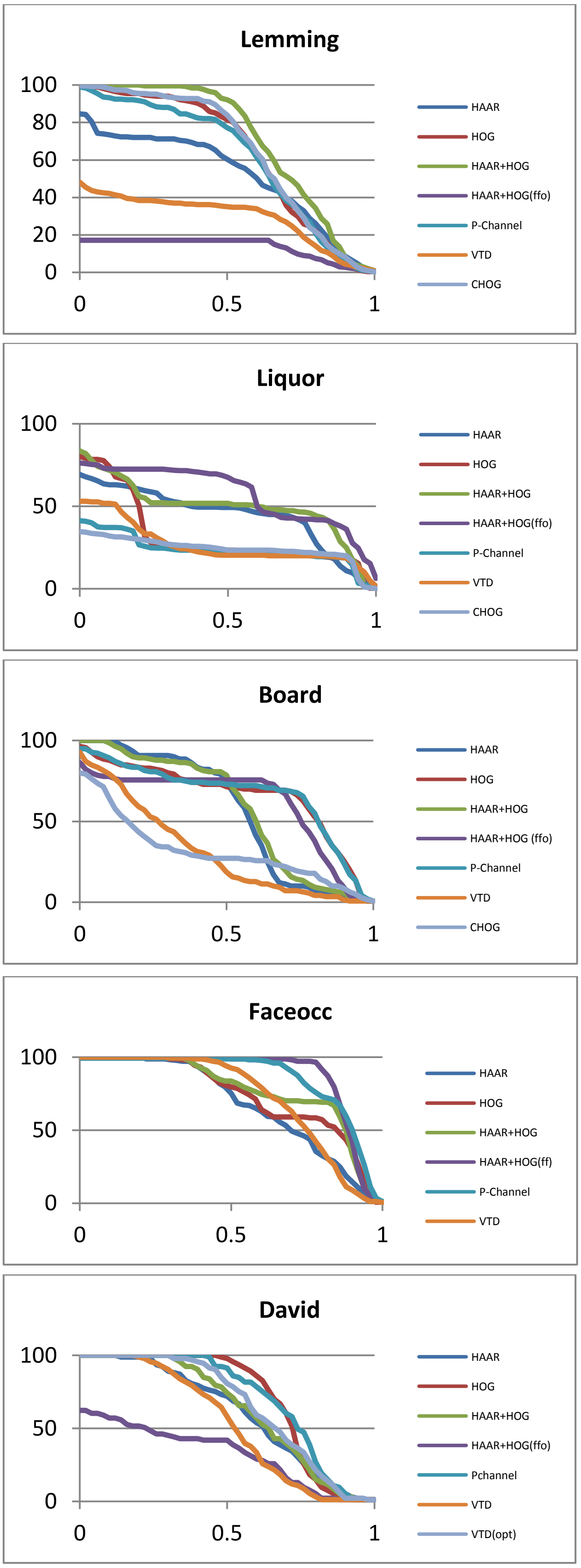

Figure 4: The sequences used in the evaluation. From top to bottom: "Lemming", "Liquor", "Board", "Faceocc" and "David"

6 CONCLUSION

In this paper, we presented OnEye, a framework for producing and broadcasting clickable videos. The system comprises a web-based video editor that allows for object tracking with user interaction in order to guarantee correct results. The time required to produce the tracks is optimized thanks to tracker fusion techniques and fast validation process based on interpolation techniques. The outputs of the editor are so-called OnEye Videos that contains the tracks of one or several objects of interest. These videos can be played in a generic player called OnEye Player that reads the tracks and transforms clicks into events. Such an event can be the opening of a mini-browser showing a webpage where the object can be purchased. In future work, we plan to further investigate our fusion strategies in order to validate the automatic choice of the best tracker.

Acknowledgement

This work has been partially funded by the BMBF VIP project OnEye under contract number 03V0007 and the European project AlterEgo under contract number 600610.

References

[1] H. Yang, L. Shao, F. Zheng, L. Wang, and Z. Song. Recent advances and trends in visual tracking: A review. Neurocomputing, 74(18):3823–3831, 2011.This is reference No. 2

[2] M. Chate, S. Amudha, V. Gohokar, et al. Object detection and tracking in video sequences. Aceee International Journal on signal & Image processing, 3(1), 2012.

[3] Q. Wang, F. Chen, W. Xu, and M.-H. Yang. An experimental comparison of online object-tracking algorithms. SPIE: Image and Signal Processing, pages 81381A–81, 2011.

[4] Y. Wu, J. Lim, and M.-H. Yang. Online object tracking: A benchmark. CVPR 2013

[5] P. Bertolino. Sensarea: An authoring tool to create accurate clickable videos. In Content-Based Multimedia Indexing (CBMI), 2012 10th International Workshop on, pages 1–4. IEEE, 2012.

[6] H. Zhong, L. Wenyin, and S. Li. Interactive tracker–a semi-automatic video object tracking and segmentation system. Microsoft Research China

[7] I. Grinias and G. Tziritas. A semi-automatic seeded region growing algorithm for video object localization and tracking. Signal Processing: Image Communication, 16(10):977–986, 2001.

[8] http://www.clikthrough.com – June 2013

[9] http://www.videoclix.tv – June 2013

[10] http://www.wirewax.com – June 2013

[11] A. Pagani, D. Stricker, and M. Felsberg. Integral p-channels for fast and robust region matching. In Image Processing (ICIP), 2009 16th IEEE International Conference on, pages 213–216. IEEE, 2009.

[12] B. Babenko, M.-H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 33(8):1619–1632, 2011.

[13] J. Kwon and K. M. Lee. Visual tracking decomposition. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pages 1269–1276. IEEE, 2010.

[14] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. Exploiting the circulant structure of tracking-by-detection with kernels. ECCV 2012

[15] PROST: Parallel Robust Online Simple Tracking - Jakob Santner, Christian Leistner, Amir Saffari, Thomas Pock und Horst Bischof – CVPR 2010

[16] Robust fragments based tracking using the integral histogram - Amit Adam, Ehud Rivlin and Ilan Shimshoni – CVPR 2006

[17] Incremental Learning for Robust Visual Tracking - David Ross, Jongwoo Lim, Ruei-Sung Lin, Ming-Hsuan Yang – IJCV 2007