A Tool to Support the Creation of Datasets of Tampered Videos

Edoardo Ardizzone and Giuseppe Mazzola(✉)

Dipartimento di Ingegneria Chimica, Gestionale, Informatica, Meccanica (DICGIM),
Università degli Studi di Palermo, Viale delle Scienze bd.6, 90128 Palermo, Italy
{edoardo.ardizzone,giuseppe.mazzola}@unipa.it

Abstract. Digital Video Forensics is getting a growing interest from the Multimedia research community, as the need for methods to validate the authenticity of a video content is increasing with the number of videos freely available to the digital users. Unlike Digital Image Forensics, to our knowledge, there are not standard datasets to test video forgery detection techniques. In this paper we present a new tool to support the users in creating datasets of tampered videos. We furthermore present our own dataset and we discuss some remarks about how to create forgeries difficult to be detected by an observer, to the naked eye.

Keywords: Copy move forgery · Video forensics · Object tracking

1 Introduction

Nowadays, the widespread use of mobile devices has drastically increased the number of videos available online through several web channels, e.g. Flickr, Facebook, Twitter, YouTube, Vimeo, Dailymotion. The huge amount of available videos improved the free flow of information around the world, but raised the problem of the validation of such information. The content of a video may be altered either with funny purposes, or for malicious goals, e.g. to modify the evidences of a legal process, to support a political campaign or to emphasize a scoop of a TV news.

Unlike the image processing tools, which have been very popular in the digital users' community for a long time, video processing tools need more skill, above all if the goal is to alter the content of the video by deleting or adding objects to the scenes. Furthermore, in scientific literature, to our knowledge, there are not standard datasets to test the ability of Digital Forensics techniques in detecting tampered videos.

In this paper we present a tool to support digital users in creating tampered digital videos, in particular to clone objects from a video sequence to another, or to the same, video sequence. The goal is to build a dataset of tampered videos, which can be used by the Digital Forensics community to test their video tampering detection techniques. The paper is organized as follows: in section 2 we discuss some state of the art methods about Video Forensics; in section 3 we present our method for cloning objects in a video; in section 4 we present our dataset and remark some points and of our experimental tests; a conclusive section ends the paper.

V. Murino and E. Puppo (Eds.): ICIAP 2015, Part II, LNCS 9280, pp. 665–675, 2015.
DOI: 10.1007/978-3-319-23234-8_61

2 State of the Art

Multimedia Forensics is a relatively new branch of the Multimedia Processing research field, and focuses on verifying the authenticity or detecting the source of a multimedia file. While Digital Image Forensics has been widely explored in the last ten years[1], Video Forensics issues have been rather less studied [2,3]. Regarding Video Forgery Detection techniques, they can be subdivided into active and passive approaches. Active approaches [4] exploit superimposed information, as watermarks or signatures, to verify the integrity of a video file. Passive approaches use internal features to detect if a video has been tampered in some ways. Some methods propose to use noise characteristics [5] to detect possible forgeries. Other works try to detect proofs of the evidence of a double compression [6,7]. Wang and Farid [8] proposed a method to detect duplicated frames used to remove people or objects from a video.

Video forgeries may be classified [9] into: spatial domain alterations; temporal domain alterations; spatio-temporal domain alterations. Spatial alterations modify the pixels of one or more frames of a video sequence, without modifying its duration. Temporal alterations change the video sequence duration. This is the case of frame dropping or frame repetition techniques [10,11], used typically to alter the content of surveillance videos. Spatio-temporal alterations combine both of them.

Tampering methods for video sequences may be further classified into inter-frame, intra-frame and inter-video. Intra-frame forgeries are duplication of a part of a frame into the same frame and are very similar to image copy-move forgeries. Inter-frame forgeries are duplication of a part of a video into another part of the same video. Inter-video forgeries are obtained merging the content of two different videos.

In our previous works we dealt with the problem of identifying copy move forgeries in still images [12-15]. In this work we present a tool which is able to support a user in the creation of all these types of alterations (intra-frame, inter-frame and inter-video forgeries).

3 Proposed Method

The whole cloning process can be subdivided into several steps, as shown in fig. 1: Selection, Tracking, Transformation, and Blending.

3.1 Selection

The first step is the only one in which the user intervention is required. The system requires the user to select, from a frame of the Source Video, a Region of Interest (ROI), to choose the object to be cloned (fig. 2.1). The system automatically extract the centroid of the input mask, as the average value of the points of its bounding box, which will be used into the blending phase.

The system requires also a destination point into the Destination Video, where (coordinates) and when (the frame) the selected object has to be pasted. The object can be cloned into the same frame of the same video (Intra-frame), into another frame of the same video (Inter-frame), or into another video (Inter-video) .

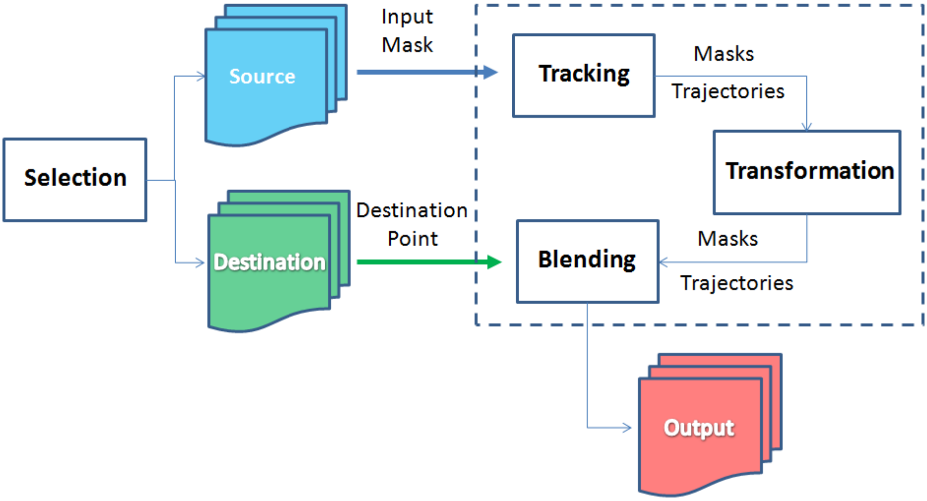


Fig. 1. Schema of the proposed method

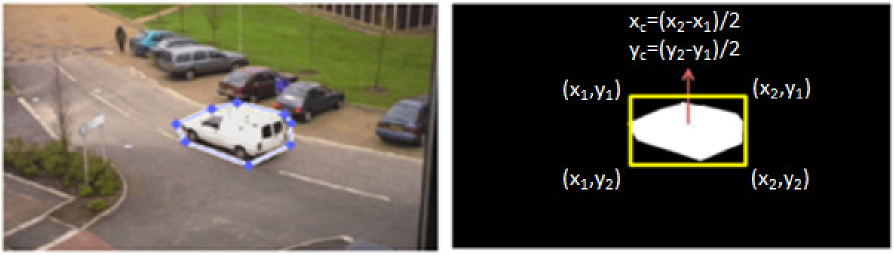


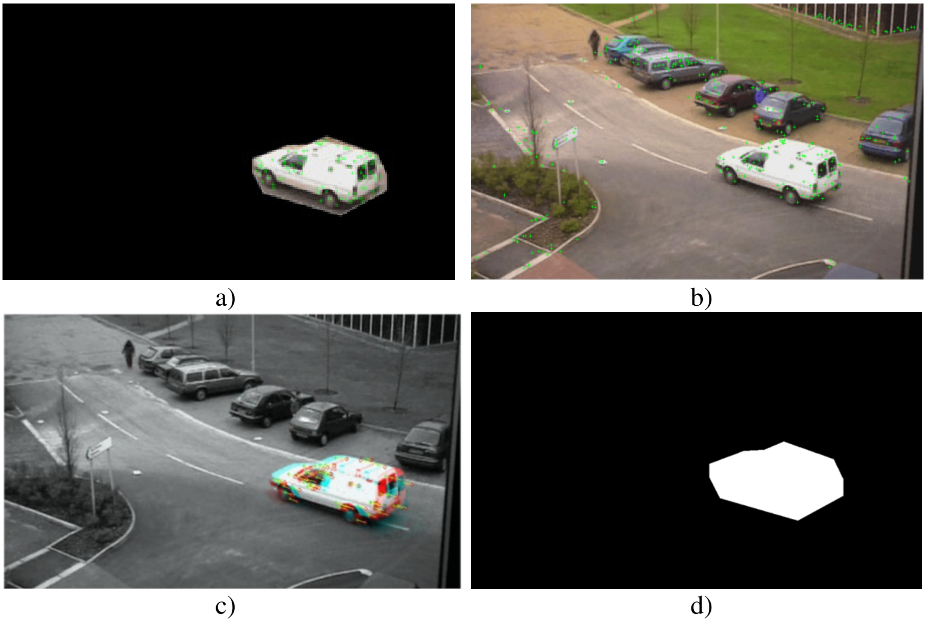
Fig. 2. Object selection and centroid extraction

This is the most tricky part of the whole process, as a good result depends on the skill of the user. In fact, the tracking algorithm performs better if the starting mask is properly selected, depending on the object visual content. Furthermore, the choice of a proper destination point may limit the presence of artifacts when pasting the selected object. These issues will be further discussed in the next sections.

The user has to select also the maximum number of frames along which the input object has to be tracked, and copied into the destination video.

3.2 Tracking

Our tracking method is based on the SURF [16] extraction algorithm, as its ability to describe the local properties of an object well fits with our goals. After the ROI R_P is selected in the previous step from the source video, we extract the SURF points



a)

b)

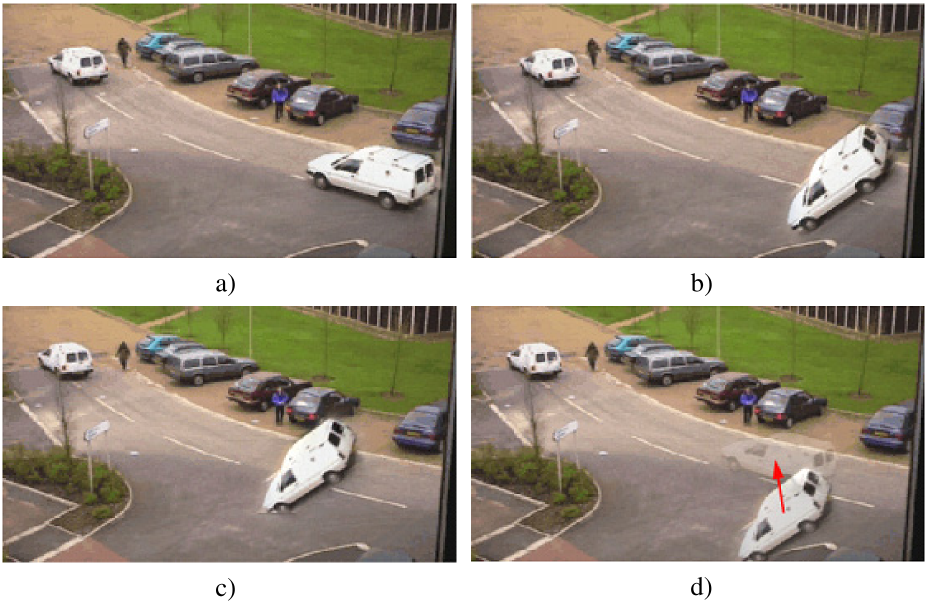
c)

d)

Fig. 3. Tracking process. Input object into R_P of the Previous Frame F_P, with superimposed points (a). Next frame F_N with superimposed matching points (b). Image matching (c). The new ROI R_N (d).

from the whole starting frame F_P, where the object to be cloned is. We use R_P to filter all the SURF points of the selected object. These, and only these, points are matched to all the points of the next frame F_N, by using the related SURF descriptors. We then select only the matching points, which all probably belong to the input object, and try to estimate the geometrical transformation T between the points into R_P, and the matching points in F_N, using the RANSAC [17] algorithm (which furthermore filters out the outliers). If enough points match, and T has been estimated, we apply it to the R_P vertices, obtaining a new ROI from the next frame R_N. In the next step of the tracking algorithm, R_N turn into the new R_P, and the F_N into F_P, and the whole process is repeated until the object quits the scene, or if it is heavily occluded by some other objects in the scene, or if the maximum number of frames is reached.

Note that we could have choosen the SIFT [18] algorithm, instead of the SURF one, as it has been widely shown that it is more robust to the geometrical transformations. We decided to use SURF for two reasons. First, SURF is less computational expensive than SIFT, and the execution time is strongly reduced. Second, when considering an object into two consecutive frames, there are very small differences in terms of its geometrical aspect. In this case, the matching process between SURF points is very robust against the distortion introduced by the object movement.


a)

b)

c)

d)

Fig. 4. Handling the trajectory of the copied object in different ways. Copied object without transformation (a). Object rotated by 50° counterclockwise (b). Object position after 10 frames, if the original trajectory is kept (c). Object position after 10 frames, if the rotation is applied also to its trajectory (d).

3.3 Transformation

In the simplest case, the selected object is pasted into the destination frame as it is. Optionally, users may apply some transformations to the selected object, to adapt the cloning to the destination scene or to create more complex forgeries.

Two different types of transformations may be applied: Geometrical; Luminance and Chrominance. Regarding the geometrical transformation we considered the full transformation matrix:

$$M = \begin{bmatrix} S_x * cos\theta + S_x * H_y * sin\theta & S_x * H_y * cos\theta - S_x * sin\theta & 0 \\ S_y * sin\theta + S_y * H_x * cos\theta & -S_y * H_x * sin\theta + S_y * cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

where S_x and S_y are the scaling factors, H_x and H_y the shearing factors and θ is the rotation angle w.r.t. the image plane. Note that setting S_x and/or S_y as negative values, the transformed object is flipped horizontally, vertically or both sides.

When applying a transformation, it is possible to select to modify also to the object trajectory. Two options can be selected in this step: applying the transformation only to the object, which will follow the original trajectory; modify also the trajectory of the object. In this second case, we further apply a translation matrix to the transformed image. The translation values are computed as the displacement between the

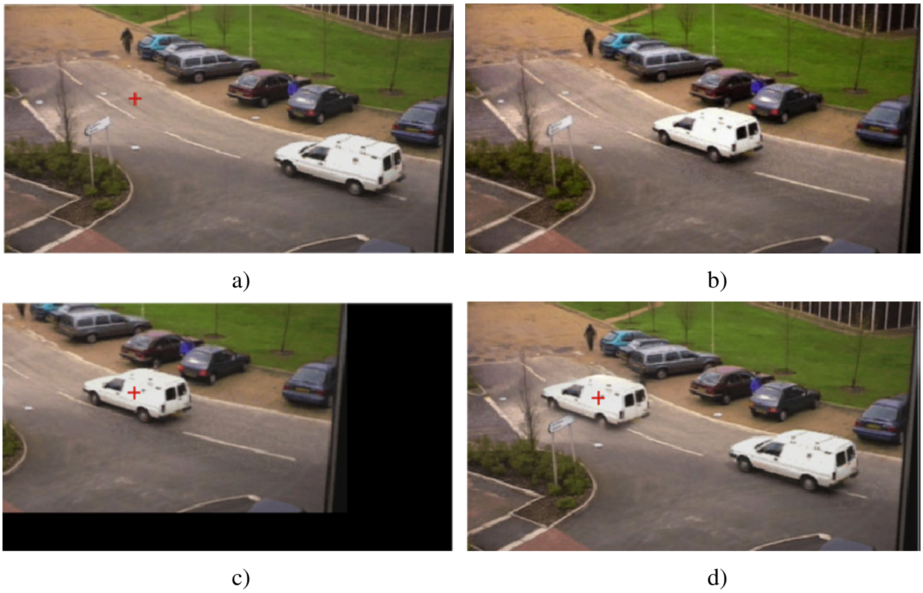

a) b) c) d)

Fig. 5. The registration process. Destination Frame with superimposed destination point (a). Source frame (b). Source frame, translated according to the offset (c). Merged frame (d)

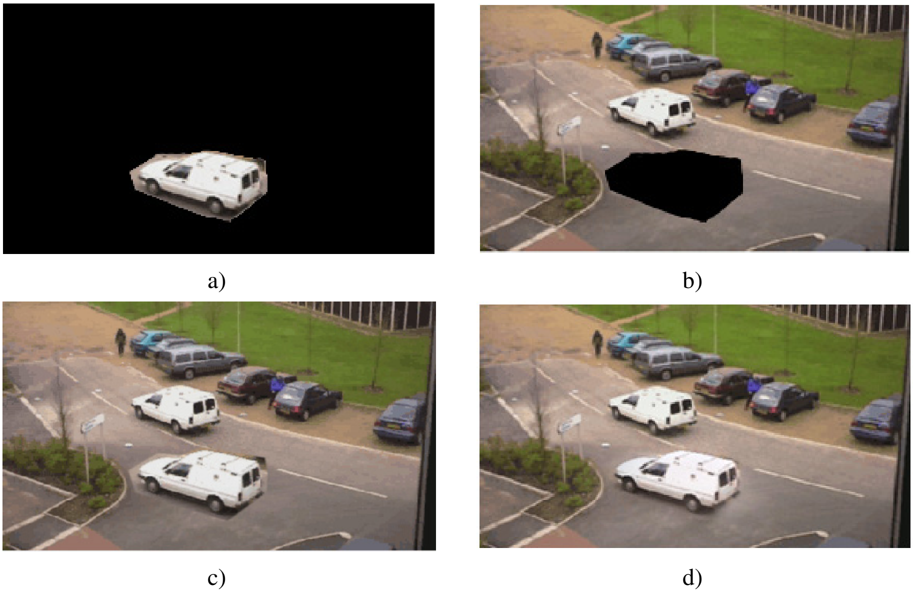

a) b) c) d)

Fig. 6. The Blending phase. The two images to blend, filtered by the ROI masks (a,b). The resulting image, without (c) and with (d) blending.

centroid of the original ROI cloned into the destination point without any transformations, and the centroid of the ROI, after the applied transformation. Note that the displacement between the original trajectory and the transformed one accumulates during each step of the cloning process.

The Luminance and Chrominance changes are allowed simply by adding values to each RGB channel. Tuning these values it is possible to change the luminance or the color of the selected object, e.g. to adapt it to the scene brightness or color tone of the destination frame.

3.4 Blending

The first part of the blending module is a registration phase. We compute the offset between the two images to be merged as the displacement between the user selected destination point and the centroid of the actual ROI, and we apply the related translation to the source image, in order to be ready for the blending phase (see fig. 5). We also register, as well, the input ROI, in order to mark out the pixels of the two images for the next blending phase.

To achieve the most natural results, the two images, after the registration step, have to be merged as best as possible (fig. 6). It is desirable that the selected object contours will be harmonized with the background of the destination frame, to avoid evident and annoying artefacts. For this goal we used the technique proposed by Hsu and Wu [19], which is based on the Laplacian Pyramid decomposition. The quality of the results depends on the number of chosen decomposition levels, above all if the brightness of the source and the destination frame are very different. On the other hand, the higher the number of levels, the higher is the execution time. Then we select, as a good tradeoff between efficiency and quality of the results, a number of levels equal to 6.

4 Experiments

The goal of this work is to create a tool to support users in cloning objects in digital videos. It is very difficult to give an objective evaluation to the results of a cloning method, as no proper metrics can be used in this case. On the other hand, using a subjective criterion to evaluate the results may be meaningless, as the quality of the output videos strongly depends on the user ability: selecting the most accurate ROI; choosing a good destination zone into which copy the cloned object, in order to make it less detectable; selecting the best transformation to adapt the object aspect to the destination scene. A subjective evaluation of the results would be an evaluation of the user cloning ability, rather than our method's potential.

Then, in this section we present our own dataset of tampered videos, which we created with the proposed tool, and which is available on demand to test video forgery detection techniques. Furthermore we will also discuss how to use our tool to obtain the best results, in terms of undetectability, to the naked eye, of the cloned areas.

Table 1. Input Video Features

VIDEO	N° Frames	Frame rate	Duration	rows	Columns
v1	172	25	6,88 s	360	640
v2	334	25	13,36 s	576	768
v3	98	25	3,92 s	540	960
v4	259	30	10,36 s	540	960
v5	554	30	18,47 s	240	320
v6	104	25	4,16 s	576	768

Table 2. The number of videos in our dataset, for each applied transformation.

Transformation	v1	v2	v3	v4	v5	v6
None	5	5	5	5	5	5
Scaling	3	2	2	2	1	5
Shearing	2	2	1	1	2	2
Rotation	2	2	1	2	5	10
Flipping	2	3	1	1	3	5
Luminance	4	4	3	1	4	4
RGB	3	3	2	2	3	5
Combination	5	5	5	5	5	5

4.1 Dataset

We created our dataset from six different videos harvested by SULFA[20] and CANTATA[21] video datasets. All the videos represent scenes of traffic control, or parking surveillance. Five of the six videos are acquired with fixed cameras, while the last one is a scene of a camera following a car along a road. Note that our implemented tracking method works, as well, with fixed and not fixed cameras, as it focuses only onto the object features, regardless of the background. We prefer to use fixed camera videos just as in these scenes the movements of the objects are more evident and, as well, cloned objects are more interesting. Tab.1 shows the principal features of the chosen reference videos. Starting from these dataset of videos, we created 160 tampered videos, with different types of cloning, as shown in table 2. Within this dataset, the average duration of a cloned slot into a destination video is of 30 frames.

Regarding the efficiency, using a Windows 7 (64 bit) machine with an Intel Core i5 2.4 GHz processor, and 4 GB RAM, the execution time is about 1,7 seconds per cloned frame then, on average, less than 1 minute for the whole process. Most of the time (65% ca) is spent in the blending phase, which is nevertheless needed to achieve good quality results.

In our dataset we created both videos with invisible and visible, to the naked eye, cloned objects, as we are interested in building a dataset with a lot of possible transformations, rather than perfect cloning results. However, on the basis of our experiments, in the next subsection we will present some suggestions to obtain forgeries that will be difficult to be detected to the naked eye.

This dataset is available at [22] (or by contacting the authors by email) to researchers who want to test their forgery detection techniques.

4.2 Remarks

In this section we present some suggestions, based on our experimental tests, to create an "invisible" cloned object:

- First of all, choosing an object full of **details**. In this case, the selected object will have a lot of interest points and the tracking method will perform better. Of course, if a homogenous area is selected, our method fails, as no SURF points can be extracted and no tracking can be performed, not even with other algorithms.
- The **object trajectory** has to be as much rigid as possible. In fact when the object changes its trajectory in a non rigid way (e.g. a car along a road with turns right or left), the number of matching points between two consecutive frames decreases (see section 3.2), the estimated transformation between the object in the two frames is less accurate, and so the relative transformed mask. Therefore the tracked object may be deformed or may have lacking portions, even if correctly tracked. The same problem occurs when the tracked object starts exiting the scene, as a lower number of interest points are extracted.
- An accurate **ROI selection** is very important. When creating the mask to select the object to be copied, the ROI polygon must be as close as possible to the object edges, to discard background information which will influence the blending phase. On the other hand, if the ROI is too close to the object boundaries, some of the interest points of the object could be not included in the mask, resulting a lower performance of the tracking algorithm.
- Selecting a good **destination point**. If the area into which we want to copy the desired object is too full of details, when pasting the object, also after the blending step, the difference between the source and the destination areas will be very evident, and the cloning more easily detectable. On the other hand, pasting the select object onto homogenous areas will create more visually convincing results.
- As well, above all in case of inter-video forgeries, the source and the destination areas should have similar **luminance** values, otherwise, in spite of the blending phase, the cloned area will be evident. Alternatively, the luminance difference may be corrected by using the luminance transformation function.

- A cloned moving object must be **consistent** to the other objects into the scene. For example, if we consider a scene with a lot of cars along a street and we decide to clone another car which crosses that street orthogonally, even if no cloning artefacts are revealed, any observer will be able to detect the forgery, as the object "behaviour" is highly suspicious.
- Respect the **perspective** rules. For example, if we apply a magnification to a cloned object and we put it backward with respect to the other objects in the scene, considering the camera position, it will be very evident that the object is a fake, as the size difference will reveal the forgery.

5 Conclusions

Digital Video Forensics can be considered still a new research field, even if digital watermarking techniques have been proposed for a long time to validate the authenticity of a video content. Nevertheless, active techniques, as well known, cannot be used in most of the real cases, then passive techniques are preferable for real applications. Regarding passive Video Forensics techniques, on the other hand, a lot of work has still to be done to solve the related problems, above all if we compare the actual results to those of the existing Image Forensics methods. It is then important for the researchers to have common and standard datasets to test their algorithm and compare their results with those of the same scientific community. With this work we aim to meet these needs and to give to the Multimedia researchers both a new tool to create their own testing videos, and a reference dataset to compare their results to those of the other researchers.

In our future works we plan to improve our tool to better support users, e.g. helping them to better select the desired object, suggesting to them the best areas into which pasting it into the destination frame, automatically adjusting brightness differences between the copied and destination areas, etc. We further plan to extend our dataset, including more videos and other different types of transformation. We are also working on a new forgery detection method that will be able to detect and localize the cloned areas of tampered videos.

Acknowledgements. The authors wish to acknowledge Mr. Giorgio Vaccaro for his collaboration during the implementation and the testing phases of this work.

References

1. Sencar, H.T., Memon, N.: Overview of State-Of-The-Art in Digital Image Forensics. Algorithms, Architectures and Information Systems Security **3**, 325–348 (2008)
2. Rocha, A., Scheirer, W., Boult, T., Goldenstein, S.: Vision of the Unseen: Current Trends and Challenges in Digital Image and Video Forensics. ACM Comput. Surv. **43**(4), 42 (2011). Article 26

3. Milani, S., Fontani, M., Bestagini, P., Barni, M., Piva, A., Tagliasacchi, M., Tubaro, S.: An Overview on Video Forensics. APSIPA Transactions on Signal and Information Processing **1**, e2 (2012). (18 pages)
4. Lee, S.J., Jung, S.H.: A survey of watermarking techniques applied to multimedia. In: Proc. IEEE Int. Symp. Industrial Electronics, vol. 1, pp. 272–277 (2001)
5. Kobayashi, M., Okabe, T., Sato, Y.: Detecting video forgeries based on noise characteristics. In: Wada, T., Huang, F., Lin, S. (eds.) PSIVT 2009. LNCS, vol. 5414, pp. 306–317. Springer, Heidelberg (2009)
6. Liao, D.D., Yang, R., Liu, H.M., et al.: Double H.264/AVC compression detection using quantized nonzero AC coefficients. In: Conference on Media Watermarking, Security, and Forensics, San Francisco, CA, vol. 7880, Article number: 78800Q (2011)
7. Sun, T., Wang, W., Jiang, X.: Exposing video forgeries by detecting MPEG double compression. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1389–1392. IEEE, March 2012
8. Wang, W., Farid, H.: Exposing digital forgeries in video by detecting duplication: In: Proc. Workshop on Multimedia & Security Int. Multimedia Conf., New York, NY, pp. 35–42 (2007)
9. Upadhyay, S., Singh, S.K.: Video Authentication: Issues and Challenges. International Journal of Computer Science Issues **9**(1), 409–418 (2012). No. 3
10. Malekesmaeili, M., Fatourechi, M., Ward, R.K.: Video copy detection using temporally informative representative images. In: Proc. International Conference on Machine Learning and Applications (ICMLA 2009), pp. 69–74, December 13–15, 2009
11. Chao, J., Jiang, X., Sun, T.: A novel video inter-frame forgery model detection scheme based on optical flow consistency. In: Shi, Y.Q., Kim, H.-J., Pérez-González, F. (eds.) IWDW 2012. LNCS, vol. 7809, pp. 267–281. Springer, Heidelberg (2013)
12. Ardizzone, E., Mazzola, G.: Detection of duplicated regions in tampered digital images by bit-plane analysis. In: Foggia, P., Sansone, C., Vento, M. (eds.) ICIAP 2009. LNCS, vol. 5716, pp. 893–901. Springer, Heidelberg (2009)
13. Ardizzone, E., Bruno, A., Mazzola, G.: Copy-move forgery detection via texture description. In: Proceedings of the 2nd ACM workshop on Multimedia in Forensics, Security and Intelligence (MiFor 2010), pp. 59–64
14. Ardizzone, E., Bruno, A., Mazzola, G.: Detecting multiple copies in tampered images. In: International Conference on Image Processing, pp. 2117–2120 (2010)
15. Ardizzone, E., Bruno, A., Mazzola, G.: Copy-move forgery detection by matching triangles of keypoints. IEEE Transactions on Information Forensics and Security (2015, in press)
16. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: speeded up robust features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006)
17. Fischler, M.A., Bolles, R.C.: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Comunications of the ACM **24**(6), 381–395 (1981)
18. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision **60**(2), 91–110 (2004)
19. Hsu, C.T., Wu, J.L.: Multiresolution Mosaic. IEEE Transactions on Consumer Electronics **42**(4), 981–990 (1996)
20. http://sulfa.cs.surrey.ac.uk/index.php
21. http://www.multitel.be/cantata/
22. http://www.dicgim.unipa.it/cvip/