The InVID Plug-in: Web Video Verification on the Browser

Denis Teyssou
AFP Medialab, Agence France-Presse
Paris, France
Denis.Teyssou@afp.com

Jean-Michel Leung
AFP Medialab, Agence France-Presse
Paris, France
Jean-Michel.Leung@epita.fr

Evlampios Apostolidis
CERTH-ITI
Thessaloniki, Greece
apostolid@iti.gr

Konstantinos Apostolidis
CERTH-ITI
Thessaloniki, Greece
kapost@iti.gr

Symeon Papadopoulos
CERTH-ITI
Thessaloniki, Greece
papadop@iti.gr

Markos Zampoglou
CERTH-ITI
Thessaloniki, Greece
markzampoglou@iti.gr

Olga Papadopoulou
CERTH-ITI
Thessaloniki, Greece
olgapapa@iti.gr

Vasileios Mezaris
CERTH-ITI
Thessaloniki, Greece
bmezaris@iti.gr

ABSTRACT

This paper presents a novel open-source browser plug-in that aims at supporting journalists and news professionals in their efforts to verify user-generated video. The plug-in, which is the result of an iterative design thinking methodology, brings together a number of sophisticated multimedia analysis components and third party services, with the goal of speeding up established verification workflows and making it easy for journalists to access the results of different services that were previously used as standalone tools. The tool has been downloaded several hundreds of times and is currently used by journalists worldwide, after being tested by Agence France-Presse (AFP) and Deutsche Welle (DW) journalists and media researchers for a few months. The tool has already helped debunk a number of fake videos.

CCS CONCEPTS

• **Information systems** → **Mashups**; *Video search*;

KEYWORDS

Video verification, User-generated content, Reverse video search, Forensics, Keyframe selection, Multimedia verification

1 INTRODUCTION

Verifying images and videos posted by eyewitnesses of an event on social networks, especially during breaking news events, or debunking "fake news", misinformation, disinformation or hoaxes, has become part of the daily routine in newsrooms. But those processes remain rudimentary, time-consuming and cumbersome for journalists; the latter have to manually generate screenshots, while watching the video, and use them to query reverse image search engines, to master several ever-changing tools, to scroll down endless social media users' timelines to find related information, copies or clues that allow to identify an eyewitness or an event, or to extract more knowledge about a media item.

Most of those skills and expertise are documented by several authoritative sources such as the Verification Handbook [33][1], or the recommendations from First Draft News[2] (a US-based nonprofit coalition aiming to provide practical and ethical guidance in how to find, verify and publish content sourced from the social Web). Nevertheless, very few tools really help journalists in their verification routines. In practice, journalists and investigators need to switch back and forth among a multitude of online tools and services, each of which addresses only a small part of the journalistic verification process.

Through design thinking methodology, observing and understanding of journalistic workflows and of the difficulties encountered by professionals when verifying information, we have released (on July 3, 2017) in "open beta" the first version of a browser plug-in, designed as a verification "Swiss army knife". The tool, which has been released as open source software[3], provides a unified view over a number of third party services and novel technologies developed within the InVID[4] and REVEAL[5] research projects, through a single graphical user interface, aiming to help journalists to get additional related information about the content that they try to verify.

2 RELATED WORK

The problem of online information verification is very complex and touches upon a number of research fields, including media studies and journalism (e.g. best journalistic practices for verifying user-generated content [33]), social network analysis (e.g. rumour spread

MuVer'17, October 27, 2017, Mountain View, CA, USA

ACM ISBN 978-1-4503-5510-0/17/10…$15.00
https://doi.org/10.1145/3132384.3132387

[1] http://verificationhandbook.com
[2] https://firstdraftnews.com
[3] https://github.com/invideu/invid-verification-plugin
[4] http://www.invid-project.eu/
[5] https://revealproject.eu/

over social networks [28]), knowledge engineering and computational fact checking [38], multimedia analysis and forensics [13, 34], and social media mining [6]. In this section, we focus on the areas that are most pertinent to the development of the presented plug-in, namely video fragmentation for keyframe selection (Section 2.2), multimedia forensics (Section 2.3) and context-based multimedia verification (Section 2.4). But we first start our discussion by presenting a number of related systems and services (Section 2.1) that are currently available on the market and try to address different aspects of the video verification problem.

2.1 Verification systems and services

A lot of tools are used in the journalistic verification process; these include search engines, online translators, video players and editing software, map services (e.g. Google maps, Street view, Bing maps, Open Street Map, Nokia Here, Yandex maps, Wikimapia.org) and other web services and applications (e.g. providing historical weather information). Journalists often rely on the YouTube DataViewer[6] of Amnesty International for performing reverse image search based on the video thumbnails, or simply take screenshots while watching the video and upload them on reverse image search services such as Google images. For still images, plug-ins like RevEye[7] or TinEye[8] (linked to the respective search engines) are also used. Yet, all those tools require experience and remain rudimentary and cumbersome to use (e.g. jumping from one tool to another to check a location or landmark). To our knowledge, there is currently no integrated solution for comprehensively addressing the verification needs when dealing with user-generated content.

2.2 Fragmentation and keyframe selection

A core operation for many video analysis applications, including video annotation, summarization and detection of near-duplicates, is the identification of the temporal structure of the video. The most common approach relies on the detection of the elementary parts of the video, called shots, which correspond to sequences of frames captured without interruption by a single camera. Several methods have been proposed to address this task (e.g. [1, 3, 11, 36, 37]), which is now considered as a solved one.

Nevertheless, when dealing with user-generated videos the shot-level fragmentation is too coarse and fails to reveal too much information about their structure, since these videos do not contain the typical video editing effects (e.g. for merging different pieces of video or adding transition effects) and are most commonly captured without interruption with the help of a single camera/smartphone, thus being single-shot videos. For this type of video a more fine-grained segmentation into sub-shots is appropriate, to identify the different visually coherent parts of the video. To this direction several sub-shot segmentation algorithms have been introduced. Most of them are related to video summarization and keyframe selection (e.g. [9, 15, 19, 27]), some of them focus on analyzing egocentric videos (e.g. [17, 24, 39]), others are used as a first step for detecting duplicates (e.g. [8]), or for supporting indexing and annotation of personal videos (e.g. [25]) or video rushes (e.g. [2, 10, 23, 29]).

Driven by the needs of media experts for verifying the integrity and authenticity of video content under time-pressure (which is the typical case when verifying content about breaking news), we built a very fast method that fragments a single-shot video into sub-shots and extracts a set of representative keyframes. The latter can be then used for assessing the originality of the video content by means of reverse image search. Details about the developed sub-shot segmentation algorithm are given in Section 4.3.

2.3 Multimedia forensics

The field of multimedia forensics focuses on methods for detecting traces of tampering in multimedia and extracting information about the history of media items using both content and metadata. Image forensics is an established field, and a number of surveys and evaluations of state-of-the-art techniques are available [4, 35, 41]. The latter include methods that try to identify whether an image has been tampered (tampering detection), attempt to deduce where the tampering has taken place (tampering localization), and try to detect other, generic and often innocuous operations that have taken place on the image, such as recompression, rescaling or global enhancements. Of the three, the most relevant for multimedia verification is tampering localization, since generic operations are often unrelated to verification, and tampering detection algorithms are not favored by experts, as they typically do not give explanations for their conclusions but operate as *black boxes* instead [42].

Video forensics is a relatively younger field compared to its image-based counterpart, and has yielded more limited success [30, 34]. Besides looking for tampered regions in frames, the fact that videos have a temporal aspect as well means that a significant amount of research is also devoted to detecting the addition or deletion of entire frames [7, 14]. However, the overall field has not progressed enough so far, to reach potential for applications. For that reason, the presented browser plug-in leverages recent advances of image forensics algorithms only.

2.4 Context-based multimedia verification

Recent research has shown that multimedia forensics are very hard to apply and largely ineffective on content that is sourced from the Web or social media platforms. This is mainly due to the fact that the provenance of such content is to a great extent unclear and that different platforms (e.g. Twitter, Facebook) tend to transform and resave multimedia content in a way that is destructive for the forensic traces of content [40]. To this end, recent research has investigated the potential of leveraging additional signals (i.e. context) about the content of interest with the goal of determining its veracity.

Seminal works in this area have empirically tested the potential of using a supervised learning approach for detecting newsworthy and credible posts in social media (mostly on Twitter) [6, 32]. In particular, different features have been considered: features related to the post (tweet) (including both general text features, e.g. n-grams, and twitter-specific ones, e.g. hashtag- and URL-based), the author/user, topic-based and network- or propagation-based features. Several works that follow a very similar approach [5, 16]

[6] https://citizenevidence.amnestyusa.org/
[7] https://chrome.google.com/webstore/detail/reveye-reverse-image-sear/keaaclcjhehbbapnphnmpiklalfhelgf
[8] https://tineye.com/

confirmed on different datasets the potential and high accuracy of fake post detection when using post- and author-based features. Although such methods have shown great potential in automatically distinguishing between credible and fake tweets, in practice end users (journalists) are often reluctant to rely on algorithmic outcomes for deciding on the veracity of online content. To this end, the presented plug-in only computes some of the *credibility features* investigated by previous works and presents them to end users without providing any automatically produced score quantifying the veracity of an input video.

3 INVID PLUG-IN DESIGN APPROACH

In the InVID project, we adopted –since the very beginning- design thinking as a methodology to better respond to the user needs and to implement a more iterative development process with end users. Apart from interviews with journalists dealing with user-generated content verification in their daily work, we also analyzed many real-life use cases of breaking news situations, where eyewitness content was playing a key role in those events' reporting. Particularly, the participation of InVID, through the consortium partner AFP, in the CrossCheck initiative launched by First Draft News on the French presidential election, was very valuable to observe the challenges faced by the teams of journalists trying to debunk rumors and fake news. This overall analysis was key to understanding the difficulties that journalists are facing and where an accurate usage of technology could help them save time and be more efficient. Several prototypes were made for different tools, such as Python scripts to trigger fast reverse image search on YouTube videos or to automate the advanced search on Twitter by time interval up to the minute. In the meantime, sophisticated multimedia analysis services such as video context analysis (Section 4.1), video keyframe selection (Section 4.3), and image forensics (Section 4.5) were integrated in the tool. While the goal of InVID is to develop a full knowledge verification platform to detect emerging stories and assess the reliability of newsworthy video files and content spread via social media, we decided to share our work with the journalistic community[9] by designing a browser plug-in wrapping up several tools in a single interface, implemented in HTML, CSS and JavaScript. The browser plug-in appeared as the best solution to combine the available tools, to engage with the community of end users and to provide also some long lasting modules (Twitter advanced search, Metadata reader, Magnifier, YouTube Thumbnails reverse search), locally in the browser, always at the user's fingertips.

4 PLUG-IN VERIFICATION MODULES

4.1 Video context analysis

This module aims to assist analysts by providing contextual information about the video, which can often be exploited for verification. Although part of the information provided by this module can be accessed already by visiting the page where the video was published (e.g. YouTube), the module isolates and aggregates verification-relevant information and presents it to the investigator

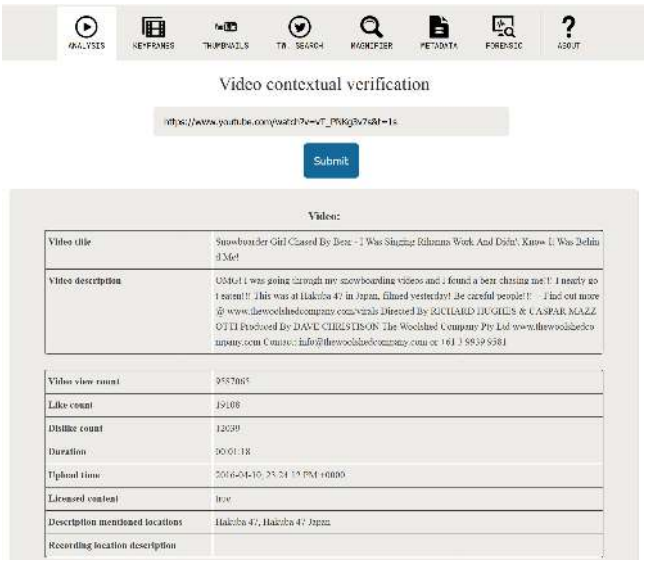

Figure 1: An example of metadata of the video contextual verification module for a fake video.

in a digestible format, organized in five categories: a) video metadata, b) channel metadata, c) comment analysis, d) external search, and e) Twitter timeline analysis.

Video and channel metadata are collected from the YouTube and Facebook APIs[10], and presented in a compact form to the investigator. Besides the video name and description, these include information such as the video and channel views count, video upload date and channel creation date (converted to GMT), plus any locations mentioned in the video description, extracted using the Named Entity Extraction functionalities of the Stanford CoreNLP library[11] (Figures 1 and 2). This contextual information is aimed at providing a first overview of the video context and to spot discrepancies between the actual metadata and the associated claims, e.g. with respect to when and where the claimed event took place versus when and where the video was uploaded, and how old, reliable, and relevant the channel appears to be.

Comment analysis aims to assist the investigator by scanning through the comments and finding the ones that are potentially related to verification. These verification-related comments are currently extracted based on a list of keywords, such as "lies", "fake", "wrong", and "confirm"[12]. Comments that contain these keywords are marked as *verification comments* and presented in a compact form. helping investigators quickly sift through them and see whether some user has already identified some information indicating that the video is real or fake. Overall, comments can be an important source for verifying videos since they offer a view on the observations of other users. At the same time, they have limitations; newly appearing videos may not contain any useful comments for some time, as users are still trying to verify the video.

[9]Media studies and media education scholars have also shown great interest in using the plug-in.

[10]In the future, more video platforms with publicly accessible API will be supported.
[11]http://stanfordnlp.github.io/CoreNLP/
[12]The list is currently under review with the goal of expanding it and also translating it to additional languages.

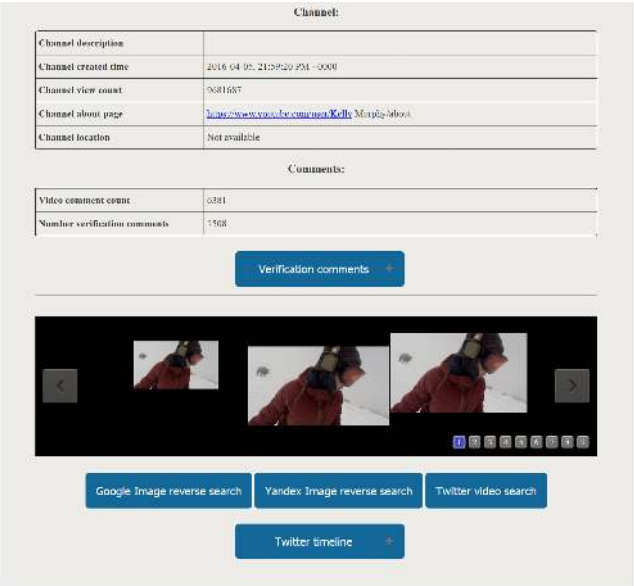

Figure 2: An example of the channel metadata, the verification comments and the external search functionality of the video contextual verification module for a fake video.

Preliminary efforts in attempting to use comment features for detecting fake videos have shown that it may take at least 6 hours to have enough comments to come to a reliable conclusion [31].

Another functionality provided by the contextual verification module is external search using third party services. A first type of such services include reverse image search platforms, such as Google and Yandex, that provide links to near-duplicate versions of a query image on the Web (if available). This aims to help the analyst find out whether the content is actually from an older event and is being reposted under a false context. To this end, the video thumbnails provided by the video platform API are sent to the respective reverse image search services. In addition, external search includes the search for posts of the input video on Twitter. The video URL is sent to Twitter as a query, and the search returns all posts sharing the video. This allows the investigator to evaluate Twitter activity around the video. Similar to other functionalities in this module, these steps could be taken by the investigator independently. However, integrating and presenting them on the same page offers a comprehensive platform that can significantly speed up the verification process.

4.2 Twitter search

Twitter is widely used by journalists to discover new information, especially during breaking news events. In the plug-in, we enhanced the Twitter advanced search by allowing the user to query this source by time interval up to the minute. This is done through automation of the conversion of regular calendar dates into Unix timestamps; a trick that some journalists were using manually until now through the Epoch Converter[13] website, which requires to copy and paste the number strings created by the aforementioned website into the Twitter search panel using the "since" and "until" operators.

This extended functionality now offers to journalists and media scholars an efficient and fast way to go back in time to the first tweets after a breaking news event or to document the Twitter coverage of past events. It also allows to quickly change, if needed, the time interval to narrow or expand the search (Figure 3).

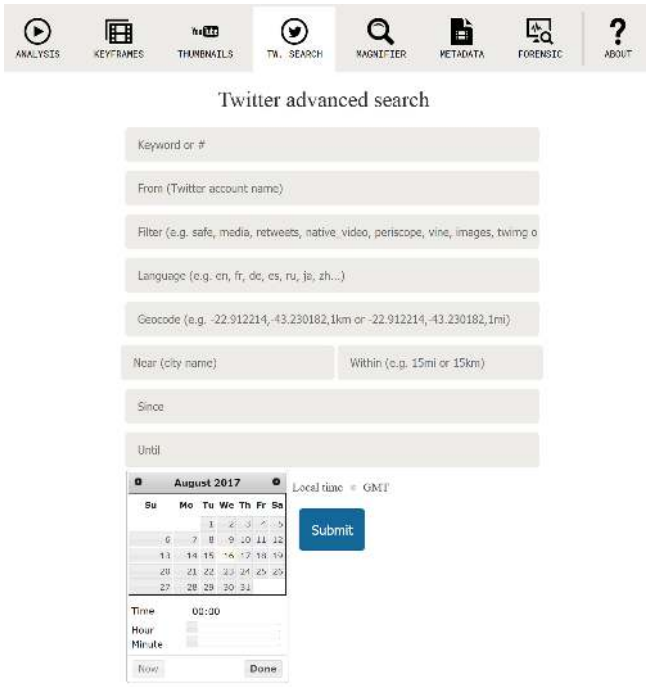

Figure 3: A view of the Twitter advanced search with the calendar (with date and minute) function.

4.3 Keyframe selection

This module selects a set of keyframes from a single-shot video by detecting visually coherent parts of it (i.e. sequences of frames having only a small and contiguous variation in their visual content) and extracting one representative keyframe from each part. The decomposition of a single-shot video into the aforementioned fragments (called sub-shots in the following) is based on the assessment of the visual resemblance of neighboring video frames with the help of the Discrete Cosine Transform (DCT), which is similar to the applied transformation when extracting the MPEG-7 Color Layout Descriptor [18]. As shown in Figure 4, the employed algorithm initially resizes each frame to $m \times m$ dimensions (step 1) and represents it as a sum of cosine functions oscillating at different frequencies via a two-dimensional DCT (step 2), forming an $m \times m$ matrix ($m = 8$ in Figure 4) where the top-left element corresponds to the DC coefficient (zero-frequency) and every other element moving from left to right and from top to bottom corresponds to an increase in the horizontal and vertical frequency by a half cycle, respectively. Following, the top-left $r \times r$ part ($r < m$) of the

[13] https://www.epochconverter.com

computed matrix ($r = 3$ in Figure 4) is kept, while high-frequency coefficients are discarded, thus removing information related to the visual details of the image (step 3). Finally, a matrix reshaping process is applied to piece together the rows of the extracted $r \times r$ sub-matrix to a single row vector (step 4), and the DC coefficient is then removed (step 5), forming a row vector of size $r^2 - 1$ that represents the image.

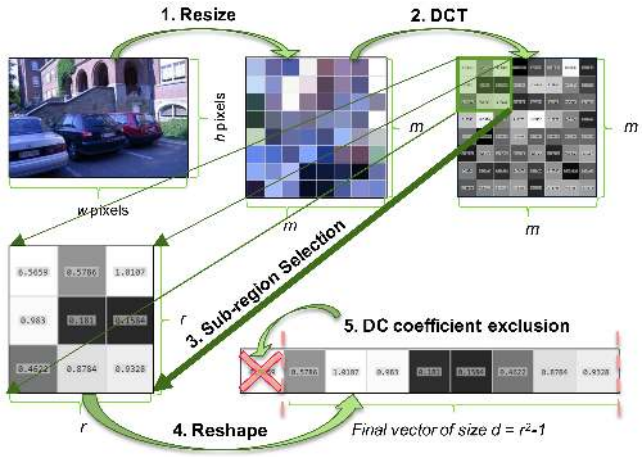

Figure 4: The steps of the applied analysis for extracting the DCT-based representation of each processed video frame.

The visual similarity between a pair of frames is estimated by computing the cosine similarity of their descriptor vectors. This process is applied for any pair of consecutively selected frames via a fixed-step sampling strategy which keeps 3 equally distant frames per second. After analyzing the entire set of selected frames the algorithm produces a series of similarity scores, which is smoothed (with the help of a sliding mean average window of size 3) for reducing the effect of sudden, short-term changes in the visual content of the video (such as the ones introduced after camera flashlights or slight hand movement of the camera holder). The turning points of the smoothed series are then identified by computing its second derivative, and each one of them signifies a change in the similarity tendency and therefore a sub-shot boundary. Through this process the algorithm indicates both sub-shots with minor or no activity, and sub-shots with gradually, but also consistently, changing visual content. As a final processing step, the representative keyframe selected for each sub-shot of the former type is its middle frame, while for the latter type of sub-shots the frame that corresponds to the point in time where the change of visual content is most pronounced is chosen. The selected keyframes are shown to the user of the toolkit, to allow performing reverse search through the Google image search engine.

4.4 Keyframe magnifier and video metadata

Implicit knowledge from a scene depicted in a video keyframe, such as car plates, banners, signs, shop names, points of interest, etc. can be used by journalists to determine whether an image or a video keyframe is really related to the location where an event is allegedly taking place. To support the inspection of keyframes, the plug-in offers a "magnifier" feature (Figure 5) based on the Elevatezoom[14] JavaScript library. We have added two algorithms that the user may trigger: one to enhance the sharpness of the image and the other, a bicubic enhancement filter, to increase the image size without harming the image readability. This functionality allows journalists, if the quality of the image allows it, to detect meaningful details within the image to confirm a location or identity, or to spot pixel incoherences that may alert of possible tampering.

Furthermore, the plug-in includes an Exif metadata reader based on an Exif JavaScript library for still images and the MP4Box.js library for video in mp4 or m4v format. Stored metadata within the image or the video, such as creation or modification date, codecs, caption if any, geocoordinates, resolution, duration, etc. are displayed in a table.

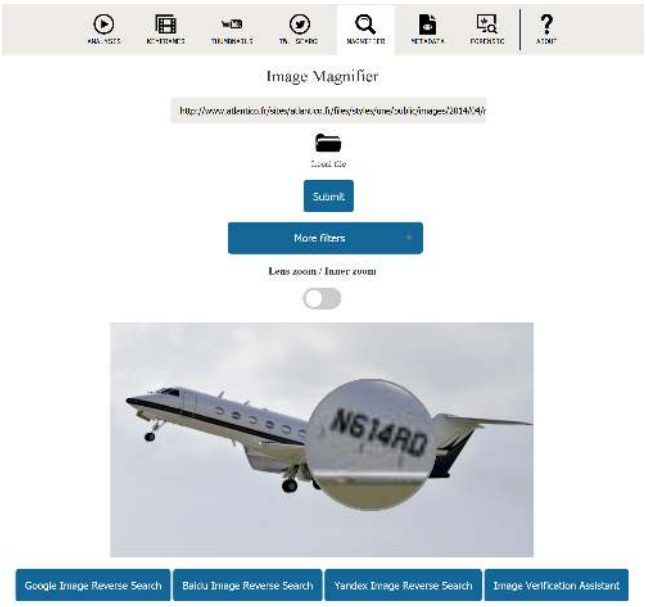

Figure 5: A view of the magnifier feature to help journalists focus on details within an image or a keyframe.

4.5 Forensic analysis of keyframes

The image forensics module provides analysis tools for tampering detection in images by incorporating a number of state-of-the-art algorithms for tampering localization. The functionalities and interface of this module, described in detail in [42], are designed to cover the needs of news professionals for news-related image verification (Figure 6).

The module exposes the results of seven state-of-the-art algorithms that process the image and return a number of localization maps. These include Double JPEG Quantization [22], JPEG Ghosts [12], JPEG Blocking Artifact Inconsistencies [21], Median Filtering Noise Residue, Discrete Wavelet High Frequency Noise Variance [26], Error Level Analysis [20], and a novel algorithm, named GRIDS

[14] http://www.elevateweb.co.uk/image-zoom

and developed within the REVEAL project[15], aiming at detecting JPEG Blocking Grid Inconsistencies.

Figure 6 shows the analysis results for a tampered image, where several of the algorithms have returned strong indications that the face may have been edited. The algorithms are chosen to cover, as widely as possible, a range of tampering traces, and are accompanied by descriptions and instructions on how to interpret the outputs, and visual examples of detections and non-detections. The aim is to assist investigators with no prior experience in image analysis, to take advantage of these tools. The service also provides a "magnifying glass" feature which allows investigators to explore details in the image or the output maps, and export their findings in an annotated PDF report that can be used for sharing their observations.

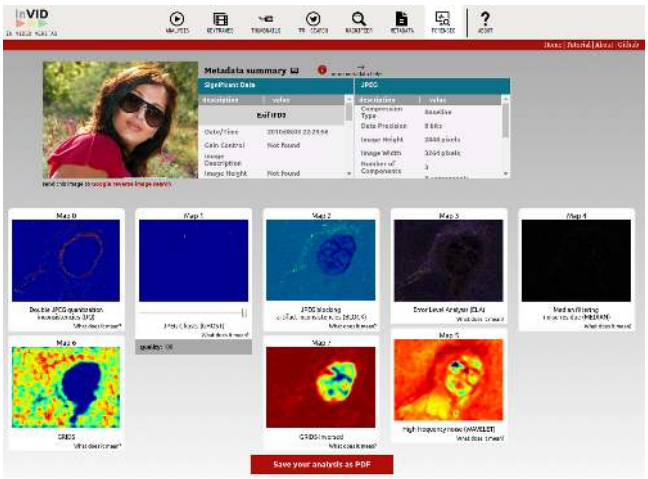

Figure 6: The results of the image forensics analysis module for a tampered image.

5 CASE STUDIES

The presented verification plug-in has been used during two months in a professional environment by a dozen journalists at Agence France-Presse (AFP) as well as by Deutsche Welle (DW) social media journalists. The plug-in allowed, e.g. to quickly debunk a fake video allegedly on the robbery of a Manila (Philippines) casino in a hotel resort on the June 2nd, 2017. The video was in fact depicting a previous robbery in a casino perpetrated in Surinam at the end of 2011. Checking the reverse image search results on Google after extracting the video keyframes, was enough to debunk this example. Through the Twitter advanced search feature of the plug-in, it is possible to easily document breaking news events from the past and to track fake images or videos that were shared on Twitter during those events. A quick search on June 19, 2017 afternoon (using a "media" parameter as filter) returned an image from an arrest in London captured in a previous terror attack. The same image reappeared on Twitter during the Champs Elysées failed attack on that particular day. Earlier, during the already mentioned CrossCheck initiative on the French presidential election, the magnifier feature allowed to prove that a jet, allegedly used by a candidate to go from one meeting to another, was in fact, registered in the United States (see Figure 5).

Last but not least, according to the analytics of the Google Chrome store, a week after the "open beta" release of the plug-in the number of total current users exceeded 400 and is now reaching 500 (Figure 7). Moreover, based on the received feedback about the tool via Twitter (indicative examples in Figure 8), other social media platforms or e-mail, the tool is currently used by journalists and experts worldwide, effectively supporting them in their efforts to debunk a number of fake videos.

Figure 7: Data about the total current users of the plug-in since its release on the Google Chrome store.

6 CONCLUSIONS

We presented the design and underlying components of a novel browser plug-in aimed at helping journalists and news professionals with the verification of user-generated video content. The plug-in, which is also available as open source software, seamlessly integrates a number of sophisticated multimedia analysis features and several third party services that are used very often within verification workflows. Through testing the plug-in in realistic settings, it has been found out that it can offer a valuable integrated tool that can considerably speed up the video verification process for journalists and media scholars, as well as non-governmental organizations dealing with video verification. The plug-in will be updated in the next months by enhancing the current features, for instance by refining the implementation of the enhancement (magnifier) feature for keyframes, by making better use of the extracted image and video metadata, and by extending the comments analysis to multiple languages.

ACKNOWLEDGMENTS

This work was supported by EU's Horizon 2020 research and innovation programme under grant agreement H2020-687786 InVID.

REFERENCES

[1] E. Apostolidis and V. Mezaris. 2014. Fast Shot Segmentation Combining Global and Local Visual Descriptors. In *Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing*. 6583–6587. Software available at http://mklab.iti.gr/project/video-shot-segm.

[2] L. Bai, Y. Hu, S. Lao, A. F. Smeaton, and N. E. O'Connor. 2010. Automatic Summarization of Rushes Video Using Bipartite Graphs. *Multimedia Tools and Applications* 49, 1 (Aug. 2010), 63–80.

[3] L. Baraldi, C. Grana, and R. Cucchiara. 2015. *Shot and Scene Detection via Hierarchical Clustering for Re-using Broadcast Video.* Springer International Publishing, Cham, 801–811.

[4] G. K. Birajdar and V. H. Mankar. 2013. Digital Image Forgery Detection Using Passive Techniques: A Survey. *Digital Investigation* 10, 3 (2013), 226–245.

[15] https://revealproject.eu/

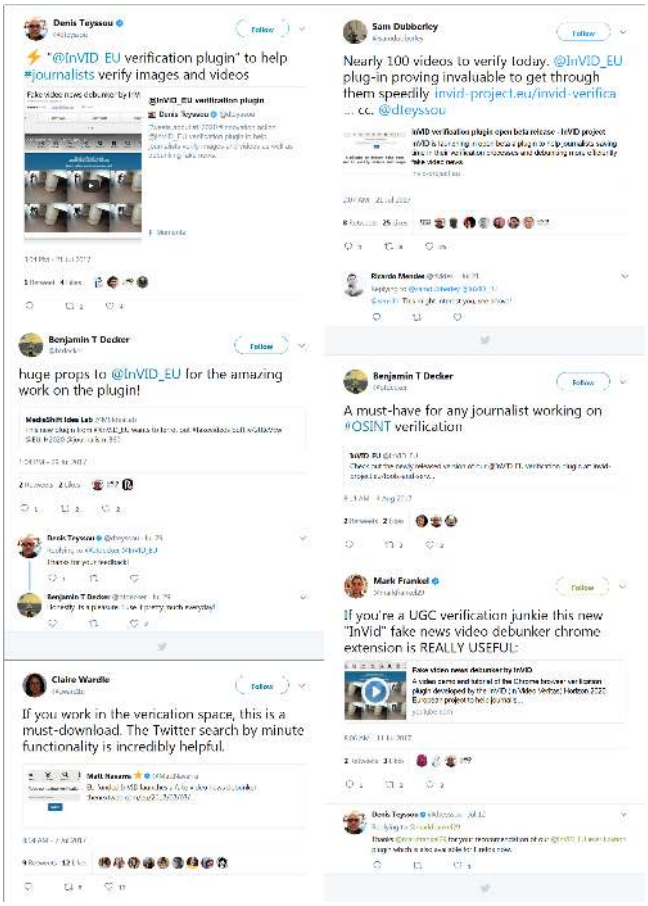

Figure 8: Indicative examples of feedback about the plug-in, shared by journalists and experts on Twitter.

[5] C. Boididou, S. Papadopoulos, L. Apostolidis, and Y. Kompatsiaris. 2017. Learning to Detect Misleading Content on Twitter. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*. ACM, 278–286.

[6] C. Castillo, M. Mendoza, and B. Poblete. 2013. Predicting Information Credibility in Time-sensitive Social Media. *Internet Research* 23, 5 (2013), 560–588.

[7] J. Chao, X. Jiang, and T. Sun. 2012. A Novel Video Inter-frame Forgery Model Detection Scheme Based on Optical Flow Consistency. In *IWDW (Lecture Notes in Computer Science)*, Yun Q. Shi, Hyoung-Joong Kim, and Fernando Pérez-González (Eds.), Vol. 7809. Springer, 267–281.

[8] W.-T. Chu, P.-C. Chuang, and J.-Y. Yu. 2010. Video Copy Detection Based on Bag of Trajectory and Two-Level Approximate Sequence Matching. In *Proceedings of the IPPR Conference on Computer Vision, Graphics, and Image Processing Conference*.

[9] S. H. Cooray, H. Bredin, L.-Q. Xu, and N. E. O'Connor. 2009. An Interactive and Multi-level Framework for Summarising User Generated Videos. In *Proceedings of the 17th ACM International Conference on Multimedia (MM '09)*. ACM, New York, NY, USA, 685–688.

[10] E. Dumont, B. Merialdo, S. Essid, W. Bailer, H. Rehatschek, D. Byrne, H. Bredin, N. E. O'Connor, G. J. F. Jones, A. F. Smeaton, M. Haller, A. Krutz, T. Sikora, and T. Piatrik. 2008. Rushes Video Summarization Using a Collaborative Approach. In *TRECVID 2008, ACM International Conference on Multimedia Information Retrieval 2008, October 27-November 01, 2008, Vancouver, BC, Canada*. Vancouver, CANADA.

[11] A. C. S. e Santos and H. Pedrini. 2016. Adaptive Video Shot Detection Improved by Fusion of Dissimilarity Measures. In *Proceedings of the 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. 002948–002953.

[12] H. Farid. 2009. Exposing Digital Forgeries from JPEG Ghosts. *IEEE Transactions on Information Forensics and Security* 4, 1 (2009), 154–160.

[13] H. Farid. 2016. *Photo Forensics*. MIT Press.

[14] A. Gironi, M. Fontani, T. Bianchi, A. Piva, and M. Barni. 2014. A Video Forensic Technique for Detecting Frame Deletion and Insertion. In *ICASSP*. IEEE, 6226–6230.

[15] I. González-Díaz, T. Martínez-Cortés, A. Gallardo-Antolín, and F. Díaz-de María. 2015. Temporal Segmentation and Keyframe Selection Methods for User-generated Video Search-based Annotation. *Expert Systems with Applications* 42, 1 (Jan. 2015), 488–502.

[16] A. Gupta, H. Lamba, P. Kumaraguru, and A. Joshi. 2013. Faking Sandy: Characterizing and Identifying Fake Images on Twitter During Hurricane Sandy. In *Proceedings of the 22nd international conference on World Wide Web*. ACM, 729–736.

[17] S. Karaman, J. Benois-Pineau, V. Dovgalecs, R. Mégret, J. Pinquier, R. André-Obrecht, Y. Gaëstel, and J.-F. Dartigues. 2014. Hierarchical Hidden Markov Model in Detecting Activities of Daily Living in Wearable Videos for Studies of Dementia. *Multimedia Tools and Applications* 69, 3 (2014), 743–771.

[18] E. Kasutani and A. Yamada. 2001. The MPEG-7 Color Layout Descriptor: A Compact Image Feature Description for High-speed Image/Video Segment Retrieval. In *Proceedings of the 2001 International Conference on Image Processing (Cat. No.01CH37205)*, Vol. 1. 674–677.

[19] P. Kelm, S. Schmiedeke, and T. Sikora. 2009. Feature-based Video Key Frame Extraction for Low Quality Video Sequences. In *10th Workshop on Image Analysis for Multimedia Interactive Services*. 25–28.

[20] N. Krawetz. 2007. A Picture's Worth... Digital Image Analysis and Forensics. Online article on: http://www.hackerfactor.com/papers/bh-usa-07-krawetz-wp.pdf. (2007). Accessed: 2016-02-26.

[21] W. Li, Y. Yuan, and N. Yu. 2009. Passive Detection of Doctored JPEG Image via Block Artifact Grid Extraction. *Signal Processing* 89, 9 (2009), 1821–1829.

[22] Z. Lin, J. He, X. Tang, and C.-K. Tang. 2009. Fast, Automatic and Fine-grained Tampered JPEG Image Detection via DCT Coefficient Analysis. *Pattern Recognition* 42, 11 (2009), 2492–2501.

[23] Y. Liu, Y. Liu, T. Ren, and K. C. C. Chan. 2008. Rushes Video Summarization Using Audio-visual Information and Sequence Alignment. In *Proceedings of the 2nd ACM TRECVid Video Summarization Workshop (TVS '08)*. ACM, New York, NY, USA, 114–118.

[24] Z. Lu and K. Grauman. 2013. Story-Driven Summarization for Egocentric Video. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '13)*. IEEE Computer Society, Washington, DC, USA, 2714–2721.

[25] J. Luo, C. Papin, and K. Costello. 2009. Towards Extracting Semantically Meaningful Key Frames from Personal Video Clips: From Humans to Computers. *IEEE Transactions on Circuits and Systems for Video Technology* 19, 2 (Feb. 2009), 289–301.

[26] B. Mahdian and S. Saic. 2009. Using Noise Inconsistencies for Blind Image Forensics. *Image and Vision Computing* 27, 10 (2009), 1497–1503.

[27] T. Mei, L.-X. Tang, J. Tang, and X.-S. Hua. 2013. Near-lossless Semantic Video Summarization and Its Applications to Video Analysis. *ACM Transactions on Multimedia Computing, Communications and Applications* 9, 3, Article 16 (July 2013), 23 pages.

[28] M. Nekovee, Y. Moreno, G. Bianconi, and M. Marsili. 2007. Theory of Rumour Spreading in Complex Social Networks. *Physica A: Statistical Mechanics and its Applications* 374, 1 (2007), 457–470.

[29] C.-M. Pan, Y.-Y. Chuang, and W. H. Hsu. 2007. NTU TRECVID-2007 Fast Rushes Summarization System. In *Proceedings of the International Workshop on TRECVID Video Summarization (TVS '07)*. ACM, New York, NY, USA, 74–78.

[30] R. C. Pandey, Sanjay K. S., and K. K. Shukla. 2016. Passive Forensics in Image and Video Using Noise Features: A Review. *Digital Investigation* 19 (2016), 1–28.

[31] O. Papadopoulou, M. Zampoglou, S. Papadopoulos, and Y. Kompatsiaris. 2017. Web Video Verification Using Contextual Cues. In *Proceedings of the 2nd International Workshop on Multimedia Forensics and Security (MFSec '17)*. ACM, New York, NY, USA, 6–10.

[32] V. Qazvinian, E. Rosengren, D. R. Radev, and Q. Mei. 2011. Rumor Has It: Identifying Misinformation in Microblogs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP '11)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 1589–1599.

[33] C. Silverman. 2014. *Verification Handbook: A Definitive Guide to Verifying Digital Content for Emergency Coverage*. European Journalism Centre.

[34] K. Sitara and B. M. Mehtre. 2016. Digital Video Tampering Detection: An Overview of Passive Techniques. *Digital Investigation* 18 (2016), 8–22.

[35] M. C. Stamm, M. Wu, and K. J. R. Liu. 2013. Information Forensics: An Overview of the First Decade. *IEEE Access* 1 (2013), 167–200.

[36] S. Tippaya, S. Sitjongsataporn, T. Tan, K. Chamnongthai, and M. Khan. 2015. Video Shot Boundary Detection Based on Candidate Segment Selection and Transition Pattern Analysis. In *Proceedings of the 2015 IEEE International Conference on Digital Signal Processing (DSP)*. 1025–1029.

[37] M. Verma and B. Raman. 2017. *A Hierarchical Shot Boundary Detection Algorithm Using Global and Local Features*. Springer Singapore, Singapore, 389–397.

[38] A. Vlachos and S. Riedel. 2014. Fact Checking: Task Definition and Dataset Construction. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*. 18–22.

[39] J. Xu, L. Mukherjee, Y. Li, J. Warner, J. M. Rehg, and V. Singh. 2015. Gaze-enabled Egocentric Video Summarization via Constrained Submodular Maximization. In

Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '15). IEEE Computer Society, 2235–2244.

[40] M. Zampoglou, S. Papadopoulos, and Y. Kompatsiaris. 2015. Detecting Image Splicing in the Wild (Web). In *Proceedings of the 2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. IEEE, 1–6.

[41] M. Zampoglou, S. Papadopoulos, and Y. Kompatsiaris. 2017. Large-scale Evaluation of Splicing Localization Algorithms for Web Images. *Multimedia Tools and Applications* 76, 4 (Feb. 2017), 4801–4834.

[42] M. Zampoglou, S. Papadopoulos, Y. Kompatsiaris, R. Bouwmeester, and J. Spangenberg. 2016. Web and Social Media Image Forensics for News Professionals.. In *Social Media In the NewsRoom, #SMNews16@CWSM, Tenth International AAAI Conference on Web and Social Media workshops*.