Video content verification using blockchain technology

Viacheslav Voronin
Moscow State University of Technology «STANKIN»
Moscow, Russian Federation
voronin_sl@mail.ru

Evgenii Semenishchev
Moscow State University of Technology «STANKIN»
Moscow, Russian Federation
sea.sea@mail.ru

Aleksandr Zelensky
Pro-rector for Research Work and R&D Politics
Moscow State University of Technology «STANKIN»
Moscow, Russian Federation
science@stankin.ru

Iliya Svirin
CJSC Nordavind, Moscow
Moscow, Russian Federation
i.svirin@nordavind.ru

Andrey Alepko
Dept. of Radio-Electronics Systems
Don State Technical University
Rostov-on-Don, Russian Federation
alepko@sssu.ru

In distributed high-performance systems that allow processing large data sets, the important task is the analysis and verification of the video data. The problem of confirming data is relevant for many areas. The need for this arises when fixing offences, banking, remote management, confirmation of actions, etc. In this paper, we describe an approach to verifying video received by a mobile device like mobile phone, tablet or PC, equipped with a camera and controlled by the operating system (Windows, Android or iOS). The proposed algorithm uses the procedure of entering the Swype code using the movement mobile camera. To improving the accuracy, we use additional information from different mobile sensors like accelerometer, gyro, barometer, and GPS. In the process of data verification, video transmission to the server is not perform what ensures the privacy of the captured video data. The server and mobile device stores data about the file size, the date of its recording, the time, the data device and its position.

Keywords— verification data, Swype code, mobile device, PROVER, blockchain

I. INTRODUCTION

The revolution of smartphones, which started ten years ago with the advent of the iPhone and tablet computers, had a significant impact on the ways users communicate in the network. Everyone suddenly became the creator and distributor of content, causing, as a consequence, his exponential growth. Such data has a large dimension. They can be multi-layered and contain data on the gradient of temperature, 3D, stickers and explanations, and so on. Big data requires large calculations, which are most often impossible on a mobile device. To analyze such data it is necessary to use Smart Cloud.

This has significantly changed and continues to change many sectors of the economy, launching the processes of digitizing the world around us. Photo and video content are actively used not only for entertainment and educational purposes but also for other needs, including economic and legal nature - financial, insurance, judicial, medical and other services. In this regard, there is a serious need for an independent, decentralized service that objectively guarantees the authenticity of the created video content and protects it from possible forgery and unfair editing.

The authenticity of digital video recordings of events and facts of commercial and legal value is often questionable because video files can be edited for forgery purposes, by using a virtual camera (emulator). Attributes such as the date of the video could also be artificially altered.

In this connection, the actual task is to operate on verification data. Verification will guarantee the originality of video data, confirm the time, location and duration of the video.

II. SOLUTIONS FOR VIDEO VERIFICATION

The authenticity of the content and the rights to it (copyright) today are mostly being approved at the level of platforms for storing and demonstrating video materials. An example of such a solution is Google content ID, which allows only to confirm the time of downloading a video on YouTube, this is the basis for the assumption of its originality, based on the principle of presumption of authorship (a person is considered as an author until the real author disputes this fact).

The drawback of this and similar solutions is that they do not allow to restore the time of real video recording, its originality, and integrity. Also, these solutions work within their platforms "only on YouTube," are provided manually by service administrators upon application and always depend on the opinion of the service administrators. That is, there is always a place for the human factor - subjectivity or simply error [1, 2].

The authors try to protect the content by watermarks and logos on the video screen, but this can only help them in the subsequent contestation of its authorship, but not to prove the fact of forgery. That means that in legal and financial matters this approach is completely inapplicable.

At the present, the blockchain community already has various online electronic notary services, which make possible to certify the existence of "Proof of existence" and the authorship of "Proof of ownership" of all kinds of files and documents and digital content:

- Block Notary - a service that helps to create "Proof of existence" of any content (photos, files, any media) using the TestNet3 or Bitcoin network. The frontend system is a mobile application for iOS that registers a document hash in a blockchain.
- Emercoin DPO Anti-fake - technology based on the Emer platform allows to create the item (product) a unique digital passport stored in a decentralized database - blockchain and provides services for

978-1-5386-8000-1/18/$31.00 ©2018 IEEE
DOI 10.1109/SmartCloud.2018.00042

managing this passport. It is focused mainly on the offline segment - it helps to register individual details (VIN, IMEI numbers) in the system to the shield of real goods from fraud.

- Stampery - blockchain technology that can verify e-mail or any files. This simplifies the process of verifying letters by simply sending them by e-mail to a specially created mailing address for each customer. Law firms use Stampery technology for a very cost-effective way of document verification.
- https://www.ascribe.io/ - service for registration of copyright, further control and distribution of digital content. It is positioned for digital works of art. It offers to register a work, put it on sale in a secure marketplace, and then monitor its use (demonstration).
- https://letsnotar.me/ - an easy service that automatically stores to blockchain a hash of files that were uploaded to it. It could launch on a smartphone, allows to get access the camera, take photos and videos and to hash them. However, it cannot guarantee that the video is recorded from a real, not a virtual camera - so it does not protect against forgery.

All these services unite one thing: they can assure a hash of a file that has already been written before or supposedly written down from the camera of the device, but they cannot guarantee its originality, integrity, and authenticity of the video. They do not protect against forgery and unfair editing because they do not have the appropriate technology to verify the video content being created [3].

Everyone can record video at any time. It is fast, simple, available and convenient. Many companies use video content in their business processes – it gives them a competitive advantage. But technologies allow others to edit and fake videos, which seriously undermines the credibility of that format.

In this work we use PROVER technology allows guaranteeing that this content was created at a specific time and place, from the camera of a device, to ensure that there are no signs of forgery and editing. The PROVER can become a functional addition, extending the capabilities of the services discussed above and the level of trust to them. Having the opportunity to 100% guarantee the authenticity of the created video file, these services will, for example, provide services for notarization of the authenticity of video statements.

PROVER technology use cases:

- Fintech, when it comes to authentication of loaner's identity. Banks and financial services providers can verify clients during the customer onboarding procedure with lower risks of identity theft. Clients can perform remote actions within the system.
- Auto insurance, dealing with fraud of security addition in contractual terms. Clients can record video evidence in case of insurance loss for insurance on demand and car sharing services. Insurance companies can receive video evidence of performed services covered by insurance (medical, repairs).
- Simple video proof of ownership, working for bloggers as example. Individuals and organizations can store a timestamped video hash on Blockchain as a digital proof of ownership for original and authentic video content.
- Public statements, to keep secure what is said before editing. Public speakers, celebrities and businessmen preventing reputational damage from montage, CGI and rapidly growing sophisticated machine learning algorithms and tools able to edit or generate fake video statements.
- Crowdsourced media platforms, to keep ownership of actual content makers. Public and crowdsourced news platforms can validate the authenticity, exclusivity and timing of video news submitted by individual contributors.
- Video platforms with user-generated content, of exclusive contributions. Both users and platforms can prove the authenticity and exclusivity of user generated video content and share monetization proceeds.
- Online dating, keeping users from fake content. Users can be sure that they are chatting with a real person on video dating websites and services.
- Outsourced reporting, carrying out remote inspection of actual performance. Employers and contractors can exchange authentic and time stamped work reports.
- Accident reports, to proof someone's position in court using evidence. Both parties involved in a traffic accident can rely on a video recording to prove authenticity of time, date and record of the accident.
- Notary actions, allowing people verify video content without visiting special parties. Parties can maintain a Blockchain video database of trusted "handshake" agreements.
- Home education and exams control, allowing remote authentication of online courses. Video recordings can be used to confirm authenticity of a person taking an online exam.

III. THE TECHNOLOGY OF PROVER

The developed service contains the following components (www.prover.io):

- A mobile app that installs on a smartphone and launches with the camera turned on or initiates the launch of the camera itself.
- A set of algorithms and utilities for integrating PROVER technology into third-party solutions and services.
- Smart Contract PROOF (only for implementation based on the Ethereum platform).

The presented technology allows performing the following actions:

- Video footage is produced by a real video camera integrated into the mobile device, and not emulated by a virtual video camera;
- The video material is complete, not edited, without gluing and insertions;
- The record was made in a certain period.

To verify the data, two approaches are used, based on changes in sensor-fixed mobile devices and analysis of video data received by the camera. The first approach uses data obtained with the help of an accelerometer, gyro, barometer, and GPS. The second approach is based on preliminary identification with visual Swype code.

The algorithm of verification with Swype code

the classical Swype code, which is being entered by moving user's finger on the touchscreen, forming a continuous line connecting the points, shown on the screen, in PROVER technology, the Swype code is being entered by moving the smartphone with the camera in a record mode.

On Figure 1, we show an example to enter Swype code using video stream from the mobile camera.

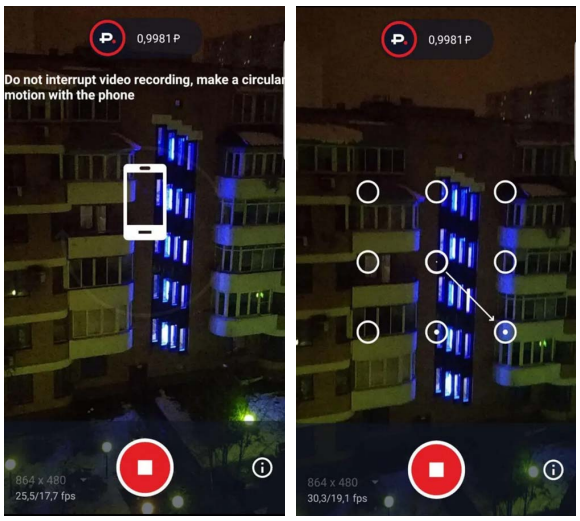

Figure 1. Graphical representation to enter Swype code.

On Figure 2 shows the algorithm for search verification using Swype code.

The following steps realize the algorithm presented in Figure 2:

Step 1: Receiving video data. The video data is transferred in parallel to the screen of the mobile device and the input of the data search and verification program. Video analysis is carried out according to the location of objects in the frame. The algorithm of search is always looking for a condition for launching verification. This condition is the circular movement of the mobile device.

Step 2: Search for frame offsets. This section of the algorithm is based on the application of the phase correlation method. This method is described in the paper [4], end realized by the following steps:

2.1) For two frames $f_1(x_1, y_1)$ and $f_2(x_2, y_2)$, we find the Fourier transformation $F_1(u_1, v_1)$ and $F_2(u_2, v_2)$ [4-7].

2.2) The cross-phase spectrum of the two spectral functions $F_1(u_1, v_1)$ and $F_2(u_2, v_2)$ stands for the ratio:

$$R_{F_1F_2} = F_1(u_1, v_1)\overline{F_2(u_2, v_2)} \Big/ \left|F_1(u_1, v_1)\overline{F_2(u_2, v_2)}\right| \quad (1)$$

The resulting expression is a spectral function with a unit modulus whose phase is equal to the phase difference of the functions.

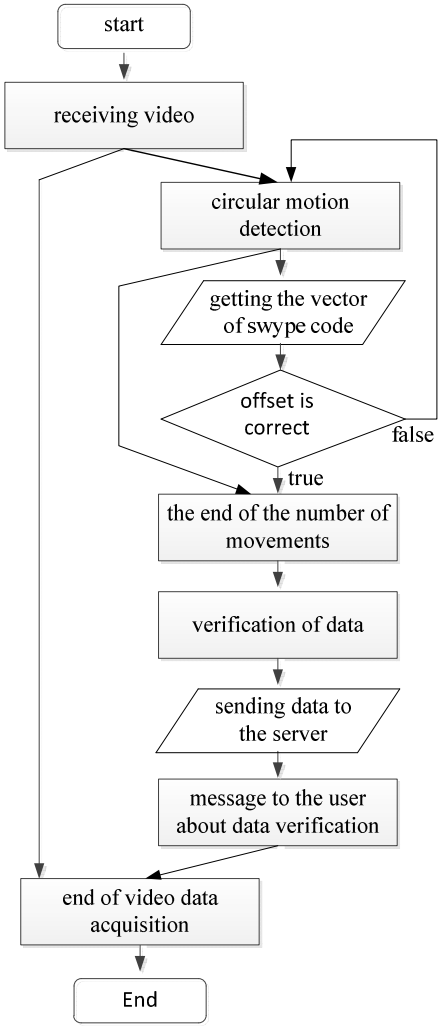

Figure 2. The verification algorithm with Swype code.

2.3) We perform the inverse Fourier transform, which is the phase correlation. Since on adjacent frames, there are the same elements up to bias:

$$F_2(u_2, v_2) = e^{-i2\pi\left(\frac{u_2 a}{x_{size}} + \frac{v_2 b}{y_{size}}\right)} F_1(u_1, v_1) \quad (2)$$

In this expression, a, b are the peaks of the delta function.

2.4) In the case of the similarity of the frames, peaks will be present. The height of the peak determines the degree of similarity, and the peak position corresponds to the displacement of frames relative to each other.

Step 3: Search for circular motion (Fig. 3).

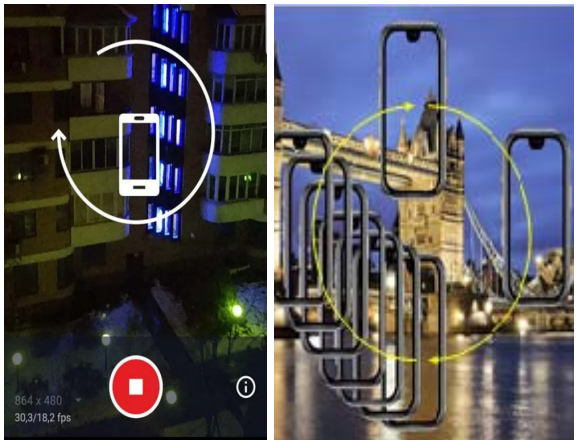

Figure 3. Determination of the circular motion.

This stage of the algorithm is performed by analyzing the displacement vectors that are in step 2. An ideal description of the change in the motion of the vectors as a function of the starting point in time for each of the axes is shown in Figure 4.

The displacement graphs relative to the 0X and 0Y axes are two harmonic functions that have a bias on $\pi/2$. Realizing that in real conditions, a person cannot describe the ideal circle. We set the confidence intervals equal to ±10% of the signal amplitude. Within the range of this interval, the result will be accepted as correct.

Step 4: Connection to the server. At this stage, there is a connection to the server. The server forwards the Swype code sequence of offsets. The sequence of movements of the mobile device relative to these elements is operation decode. An example of such a code is shown in Figure 2. A displacement of one unit within the three by three field is considered correct if the movement of the vector in the correct direction (step 2) at a time is fixed more than five steps. Each step takes place every half second. If an error occurs in the Swype code entry, the check operation is terminated. To begin verification of the data, it is necessary to repeat the action from step 3.

Step 5: Data is transferred between the server and the mobile device. The mobile device forms a data packet. This package includes information on the time of the start of the data verification, the frame size, the device description, and the file name.

Step 6: Verification. At the end of the video recording, the resulting file gets a label. Data about the end time of the survey, its size, and the hash function describing the file are transmitted to the server. Any change in the file makes a change in the information that was sent to the server.

Using this automatic algorithm for recognizing the Swype code will allow the user at the stage of video recording to be sure that afterward, later, if that the video will need to be checked for authenticity, this Swype code will be recognized by the service.

The verification algorithm using phone sensors

A stream of metadata (data from all sensors available in mobile device, with the maximum frequency - an accelerometer, a gyroscope, a magnetometer, GPS coordinates, etc.) will be recorded in parallel with the video file to prevent forgery.

To determine the position of the object in space, we introduce the global three-dimensional Cartesian coordinate system $0XYZ$ so that the axis $0Z$ coincided in direction with the direction of gravity field power lines $\vec{g}$, and the axis $0Y$ coincided with the declination of the vector of the magnetic field of the planet $\vec{h}$.

To describe the position of the sensor in the global coordinate system (GCS) we introduce a local coordinate system (LCS), whose axes will coincide with the corresponding axes of the acceleration sensors and the magnetic field. Then the position of the LCS (sensors) in the GSC can be described by four vectors:

$\vec{r}_{LCS}$ - the displacement vector of the origin of the LCS relative to the GCS origin;

$\vec{i}_{LCS}$, $\vec{j}_{LCS}$, $\vec{k}_{LCS}$ - directing vectors of the orthonormal basis of LCS expressed regarding the directing vectors of the orthonormal GCS basis.

Such description gives complete information about the orientation of the LCS in the GCS in coordinate form. The problem of determining the angular orientation reduces to finding the coordinates of the vectors $\vec{i}_{LCS}$, $\vec{j}_{LCS}$, $\vec{k}_{LCS}$ in GCS.

Because the gravitational field is more stable than the magnetic field, let us take as a basis the acceleration sensor. The acceleration sensor readings are the coordinates of the acceleration vector of the free fall, decomposed along the axes of the LCS:

$$\vec{a}_{LCS} = \{a_x, a_y, a_z\} \tag{3}$$

The normalized vector $\vec{a}_{LCS}$ is nothing else but a vector $\vec{k}_{GCS}$, i.e., the defining vector of the *OZ* GSK axis, expressed regarding the directing vectors of the LCS:

$$\vec{k}_{GCS} = \vec{a}_{LCS} \Big/ |\vec{a}_{LCS}| = \{k_x, k_y, k_z\} \tag{4}$$

The readings of the magnetic field sensor are the coordinates of the magnetic field vector, decomposed along the LCS axes:

$$\vec{m}_{LCS} = \{m_x, m_y, m_z\} \tag{5}$$

Vector $\vec{m}_{LCS}$ in the general case, it may not be parallel to the vector $\vec{k}_{GCS}$: Therefore, it needed to get its normal component:

$$\vec{n}_{LCS} = \vec{m}_{LCS} - \left(\vec{m}_{LCS} \cdot \vec{k}_{LCS}\right) \vec{k}_{LCS} \tag{6}$$

The normalized vector $\vec{n}_{LCS}$ is nothing else but a vector $\vec{j}_{GCS}$, i.e., The defining axis vector *OY* GCS, expressed through the directing vectors of the LCS:

$$\vec{j}_{GCS} = \vec{n}_{LCS} \Big/ |\vec{n}_{LCS}| = \{j_x, j_y, j_z\} \tag{7}$$

The defining axis vector *OX* GCS, expressed through the directing vectors of the LCS, is found using the vector product:

$$\vec{j}_{GCS} = \vec{j}_{GCS} \cdot \vec{k}_{GCS} = \{i_x, i_y, i_z\} \tag{8}$$

We compose the matrix of the row represented by the vectors $\vec{i}_{LCS}$, $\vec{j}_{LCS}$, $\vec{k}_{LCS}$ then transpose it and expand it into vectors (also in rows):

$$\left|i_x i_y i_z j_x j_y j_z k_x k_y k_z\right|^T = \left|i_x j_x k_x i_y j_y k_y i_z j_z k_z\right| \tag{9}$$

Thus, the vectors:

$$\vec{i}_{LCS} = \{i_x j_x k_x\}$$

$$\vec{j}_{LCS} = \{i_y j_y k_y\} \tag{10}$$

$$\vec{k}_{LCS} = \{i_z j_z k_z\}$$

determining the axis of the LCS in the GCS, in other words, determine the orientation of the LCS in the GCS.

Let's express the vector $\vec{a}_{LCS}$ in GCS:

$$\vec{a} = a_x \cdot \vec{i}_{LCS} + a_y \cdot \vec{j}_{LCS} + a_z \cdot \vec{k}_{LCS} = \{a'_{x'}, a'_{y'}, a'_{z'}\} \tag{11}$$

Vector readings are given in the quanta of the ADC of the acceleration sensor, for further calculations we translate them into the International System of Quantities (ISQ) and write the instantaneous acceleration vector:

$$\vec{a}_{phys} = \frac{range}{N} \cdot \vec{a} \tag{12}$$

where the "range" is the range of the analog-to-digital sensor converter, and N is the division price.

To take into account the gravitational field, it is necessary to reduce the vertical component by the value of the acceleration of gravity:

$$\vec{a}_g = \vec{a}_{phys} - \{0, 0, -g\} \tag{13}$$

Let's move from acceleration to speed:

$$\vec{v}_g = \int \vec{a}_g dt + \vec{v}_0 \approx \sum \vec{a}_g \Delta T + \vec{v}_0 \tag{14}$$

Let's move from speed to coordinates:

$$\vec{r}_g = \int \vec{v}_g dt + \vec{r}_0 \approx \sum \vec{r}_g \Delta T + \vec{r}_0 \tag{15}$$

We use model makes it possible to determine the orientation of the sensor relative to the global coordinate system, associated with the physical features of the planet and sensor readings. This method is to determine of linear the integration movement of the sensor in the global coordinate system and reconstruct the trajectory of the movement of the user's mobile device.

CONCLUSION

We describe an approach to verifying video data received by a mobile device. The proposed algorithm uses the procedure of entering the Swype code using the movement mobile camera. To improving the accuracy we use addition information from different mobile sensors like accelerometer, gyro, barometer, and GPS.

ACKNOWLEDGMENT

This work was supported by PROVER project (https://prover.io/).

REFERENCES

[1] Keller, Y., Averbuch, A., Israeli, M. “Pseudo polar-based estimation of large translations, rotations and scalings in images,” IEEE trans. on Image processing, 12-22 (2005).

[2] Sarvaiya, J. N., Patnaik, S., Kothari, K. “Image Registration Using Log Polar Transform and Phase Correlation to Recover Higher Scale” Journal of Pattern Recognition Research, vol. 7(1) (2012).

[3] Zitova, B., Flusser, J. “Image registration methods: a survey”, Elsevier Image and Vision Computing, 977-1000 (2003).

[4] Foroosh, H., Zerubia, J. B., Berthod, M. “Extension of phase correlation to subpixel registration.” IEEE trans. on image processing, vol. 11(3), 188-200 (2002).

[5] Reddy, B. S., Chatterji, B. N. “An FFT-based technique for translation, rotation, and scale-invariant image registration”, IEEE Transactions on Image Processing, vol. 5(8), 1266–1271 (1996).

[6] Xu, H., Hua, G., Zhuang, J., & Wang, S. A. “A Frequency Domain Approach to Fast and Accurate Image Registration”, in IEEE International Conf. on Information and Automation, 340-345 (2009).

[7] Samritjiarapon, O., Chitsobhuk, O. “An FFT-Based Technique and Best-first Search for Image Registration”, Communications and Information Technologies, ISCIT, 364-367 (2008).