electronics

Article

Fourier Image Watermarking: Print-Cam Application

Khadija Gourrame [1,*], Frederic Ros [2], Hassan Douzi [1], Rachid Harba [2] and Rabia Riad [3]

1 IRF-SIC, University of Ibn Zohr, Agadir 80000, Morocco; h.douzi@uiz.ac.ma
2 PRISME, University of Orleans, 45000 Orléans, France; frederic.ros@univ-orleans.fr (F.R.); rachid.harba@univ-orleans.fr (R.H.)
3 ERMAM-FPO, University of Ibn Zohr, Ouarzazate 45000, Morocco; r.riad@uiz.ac.ma
* Correspondence: khadija.gourrame@edu.uiz.ac.ma

Abstract: Digital image watermarking is an active research field since it provides protection, security, and authenticity of data. This paper presents development and implementation of a blind and robust watermarking application for ID images under a print-cam system. In the present case, the images are watermarked and printed on ID cards and then detected freehandedly with a smartphone camera. In order to design an efficient and robust image watermarking application, the attacks produced in print-cam processes, such as geometric distortions, must be resolved. Accordingly, the proposed watermarking approach is applied in the Fourier domain. Then, a frame-based projective rectification is integrated to deal with geometric distortions by using detection of Hough lines. Moreover, better robustness against print-cam watermarking attacks was achieved compared with the existing methods, and an Android application was designed and implemented based on the proposed scheme.

Keywords: image watermarking; print-cam process; projective distortions; Fourier transform; android application

Citation: Gourrame, K.; Ros, F.; Douzi, H.; Harba, R.; Riad, R. Fourier Image Watermarking: Print-Cam Application. *Electronics* **2022**, *11*, 266. https://doi.org/10.3390/electronics11020266

Academic Editor: Abdeldjalil Ouahabi

Received: 17 October 2021
Accepted: 27 November 2021
Published: 14 January 2022

1. Introduction

The fast growth of smartphones and the Internet have made it easier to store, edit, duplicate, and distribute digital multimedia contents. Digital image watermarking is an important topic for the community of technology information and for communication. In fact, it provides a security tool to ensure authorized access, facilitate the authentication of content, and prevent illegal reproduction. Image watermarking is a process with two main phases: first, the insertion phase, where a pattern of bits, as a message or signature, is embedded in a digital image in such a way that is invisible to human eye, and second, the detection phase, where the hidden signature is extracted from the tested image [1]. To evaluate the performance of watermarking techniques it is mandatory to satisfy some major requirements such as robustness, imperceptibility, and capacity. Robustness indicates how well the watermark resists attacks in the system, imperceptibility or invisibility describes how invisible the watermark is, and capacity tells how much information can be embedded in the image with a certain watermarking technique. Over the years, many schemes have been established to develop digital image watermarking. The improvement of hardware resources in computers poses fewer limitations on the watermarking implementation process, such as in print–scan systems where the watermarked image is printed and scanned for the detection phase. It is becoming less convenient to transfer images to computers before being able to share them online.

This situation interferes with the user experience of creating a print-cam system for instantly sharing captured images through smartphones [2]. The print-cam process is whereby the watermarked image is printed on paper and then detected with a phone's camera by taking a picture of the printed image freehandedly, as shown in Figure 1.

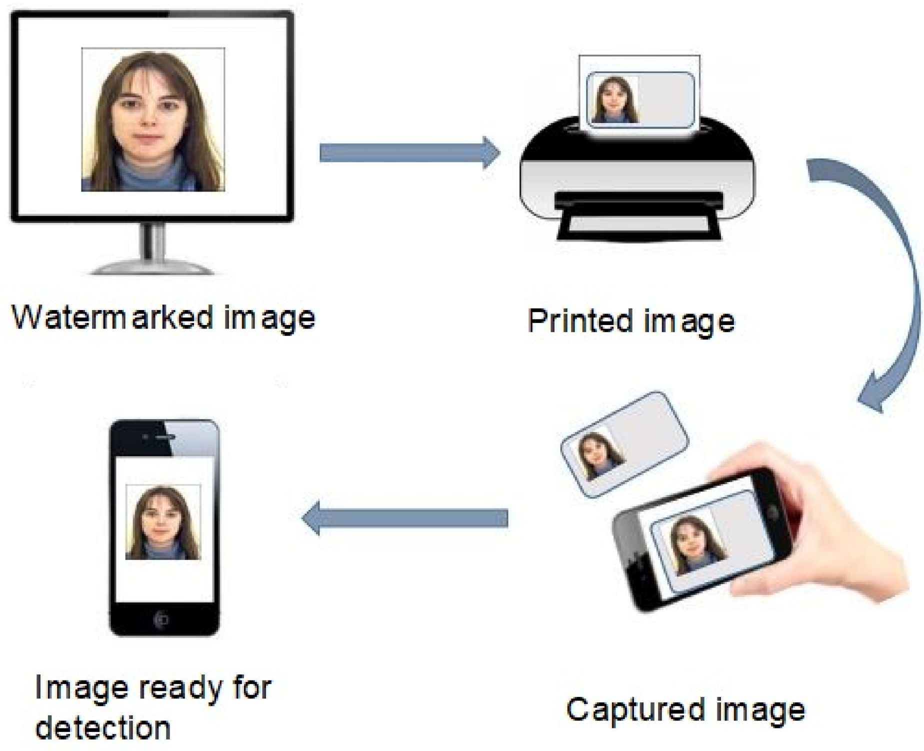


Figure 1. Print-cam process [3].

In print-cam image watermarking systems, the user takes pictures freehandedly. This produces some serious attacks or distortions that modify the captured image and hence make the watermark detection operation fail. Basically, these attacks are noise and blurring of the pixel values due to the printer and the smartphone camera, lens distortions, light reflections, and geometric distortions in the form of projective distortions that combine rotation, translation, scaling, and angular tilting of the optical axis between the camera and the object [4]. The study of robustness against geometric distortion is limited to rotation, scaling, and translation (RST) in a majority of the literature on proposed watermarking methods. This paper proposes a blind and imperceptible image watermarking method that is robust against projective deformation for freehandedly captured images. The method is developed as an android application for real-time utilization and has shown high performance in terms of robustness and time consumption.

The paper is organized as follows: Section 2 presents the state of the art related to print-cam image watermarking. Section 3 describes the proposed Fourier watermarking scheme. Section 4 details the design and implementation of the proposed watermarking application for the print-cam process. Section 5 is devoted to the conclusion.

2. Related Works

Many watermarking techniques and applications have been developed by taking into account increasing robustness, capacity, and security, along with maintaining the visual quality of the watermarked image. However, there is no existing perfect watermarking method that achieves a perfect robustness facing all watermarking attacks, as well as all kinds of geometric distortions. Overall, image watermarking techniques are carried out in the spatial domain or transformed domains [1]. Discrete cosine transform (DCT), discrete wavelet transform (DWT), and discrete Fourier transform (DFT) are commonly developed in the watermarking field. While DCT and DWT are considered to be robust against frequent signal-processing attacks such as signal filtering and compression, they are incapable of resisting geometric distortions [5]. Conversely, DFT has the potential to resist geometric deformation [6].

Geometric deformation is one of the core problems in industrial watermarking applications where the print-scan and print-cam attacks take place (print–scan is where the image is printed and then digitized with a scanner, whereas cam stands for digitization with camera). Rotation and translation (RT) deformations are, in particular, produced by the print–scan process. Print-cam systems produce higher levels of geometric attack in the form of perspective or 2D projective distortions, which include rotation, scale, and translation (RST) and the effect of the camera tilting [3,4]. This attack is the most challenging one. Geometric deformations produce a deterioration of synchronization between the extracted watermark and the embedded one. Therefore, many schemes have been developed in the literature to handle this synchronization problem. The proposed strategies can be categorized as invariant insertion methods [6–8], feature-based methods [9], template-based methods [10], autocorrelation-based methods [11], and, finally, the exhaustive search method [12]. Each method has its advantages and limitations. For example, template-based methods need a visible given shape to be inserted in the image. Exhaustive search methods need high computational costs and are not often used in practical industrial applications.

Nakamura et al. [13] and Katayama et al. [14] were the first papers to highlight the demand for print-cam image watermarking. They proposed a watermark insertion and detection technique in the spatial domain for camera-equipped cellular phones and used a frame synchronization scheme to resist geometric deformations. Their method can be robust against small geometric distortions. In addition, Takeuchi et al. [15] proposed an image-watermarking method based on frame synchronization to control geometric distortions by using guided scrambling techniques for watermark insertion. The method presented a better performance in terms of robustness against geometric lens distortion. However, it could resist only a limited amount of geometric distortion.

Pramila et al. [16] developed a print-cam watermarking method with a frame-free technique. In contrast, they proposed pseudorandom sequences and an autocorrelation function as a watermarking method in the spatial domain for colored images. In addition, the same watermarking technique is proposed with an all-in-focus imaging technique to solve the unfocused image issue in the capturing process [17]. Thongkor and Amornraksa [18] developed existing print-cam watermarking schemes for Thai ID images. The method was developed in the spatial domain, and the insertion phase was applied in the blue color channel of the picture. Instead of using a frame, they used the original image to correct the distorted image geometrically. However, the method is non-blind. Under the same conditions, Thongkor and Amornraksa [19] suggested an amelioration in the prior method using a JND model and an automatic registration method. In this paper, the imperceptibility and the robustness requirements are guaranteed with the use of a JND model and registration method, respectively. Yet, this technique keeps the requirement of the original image in the detection phase. The techniques of combining methods with watermarking insertion and detection domains are often used to enhance watermarking requirements. For example, the capacity and security requirements can be improved by using facial recognition systems [20,21] for ID image watermarking methods. Furthermore, Isa et al. [22] provide a combination of a facial recognition system and a watermarking system in the DCT domain in order to increase the security of facial recognition systems by using principal component analysis (PCA). To protect the data used in the face recognition system, they use logos and timestamps as watermarks. However, the proposed study does not provide sufficient results for robustness improvement against geometric distortions.

H. Al-Otum et al. [23] developed an image watermarking method in the spatial domain using quick respond code for smartphones, similar to the work of S. Takishita et al. [24], where they developed a watermarking method with DWT transform for 2D code, but their tests did not show resistance to projective deformations. Nam-Tuan Le [25] presented a watermarking method in the spatial domain where the projective deformations are corrected with several capture positions of the same picture.

The development of applications for image watermarking methods is increasing [23,26]. Moreover, some commercial applications have been launched, such as Digimarc [27], which introduced an application called Discover. The purpose of the application was to connect magazine users to its official websites. In print-cam systems, spatial and wavelet transform watermarking methods are the most frequently developed watermarking techniques compared to Fourier transform methods. Likewise, the watermarking methods in the Fourier domain are more robust against geometric distortions, such as translation and rotation, excluding scaling and tilting of the optical axis [28]. Delgado-Guillen et al. [29] developed a hybrid image watermarking method with Fourier and log-polar transformations for mobile platforms. Further, J. Ouyang et al. [30] proposed an advanced watermarking method based on Fourier–Mellin transform with quaternion Fourier transform. However, the testing results did not demonstrate resistance to projective distortions. Chen et al. [7,31] proposed a watermarking method for screen-cam systems, where a camera device is used to generate content displayed on screens into digital signals. The method is based on using DFT to fix the synchronization problems. Riad et al. [6,32] proposed a robust watermarking method based on Discrete Fourier Transform (DFT) for ID images. The method was developed for printed and scanned images. It presents high resistance to geometric distortions (translation and rotation) and other attacks related to the print-scan process. In the present study, the technique presented in our recent work [3] was developed and applied as an Android application for a print-cam system. The detection phase in the application combines the proposed frame-based projective registration with watermark detection.

3. Fourier Image Watermarking

In this section, we first recall the properties of the Fourier transform. Then, we explain the proposed watermarking method, composed of watermark insertion and detection steps.

3.1. Fourier Transform

The magnitude of the Fourier transform $F(u,v)$ of an image $f(x,y)$ is invariant to translation:

$$f(x-a,y-b) \leftrightarrow F(u,\ v)e^{j(au+bv)} \tag{1}$$

where a and b are translation parameters. Rotation of an image with angle θ in the spatial domain is translated to a rotation with the same angle value in the Fourier domain:

$$f(x\cos(\theta)-y\sin(\theta),x\sin(\theta)+y\cos(\theta)) \leftrightarrow F(u\cos(\theta)-v\sin(\theta),u\sin(\theta)+v\cos(\theta)). \tag{2}$$

Finally, in the spatial domain, the scaling factor a is transformed into an inverse scaling factor in the Fourier domain:

$$f(ax,ay) \leftrightarrow \frac{1}{|a|^2}F\left(\frac{u}{a},\ \frac{v}{a}\right). \tag{3}$$

The properties in (1) and (2) illustrate the natural resistance to translation and rotation (RT) of the magnitude in the Fourier transform, except for scaling changes (3). Therefore, the Fourier-transform-based watermarking method is more adapted to resist RT attacks.

3.2. Watermark Insertion

In the proposed watermarking technique, the watermark is embedded in the magnitude coefficients of the Fourier-transformed image. Only the luminance part of the color image is used in the process and the chrominance components are not modified. The block diagram of the embedding process used is shown in Figure 2.

A circular watermark W in the form of a pseudorandom vector of a binary element of length N is generated with a secret key k and inserted according to predetermined unique and optimum radius r_0 [6] in an additive way as follows:

$$M_W(x,y)\ = M_0(x,y)\ +\ \alpha \times W(x,y), \tag{4}$$

where M_W and M_0 are, respectively, the FFT magnitudes of the watermarked original images, (x, y) are the image coordinates in the Fourier domain, and α is the insertion strength factor. This last parameter is chosen based on the image quality method where the peak signal-to-ratio (PSNR) is chosen to be 40 dB, corresponding to an invisible watermark [3,6].

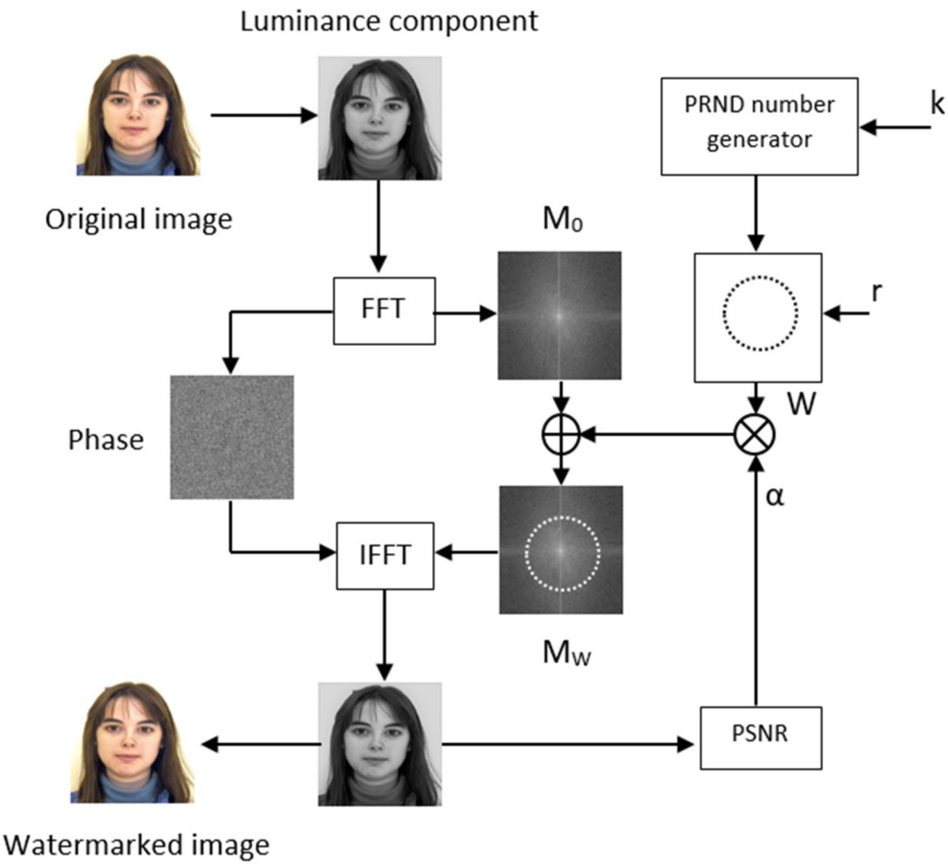


Figure 2. Fourier watermark insertion scheme.

3.3. Watermark Detection

The decoder phase performs a blind watermark detection, where the original image is not required in the detection phase. The blind decoder involves the captured image and the watermark W created with a secret key k using Pseudo Random Number (PRND) Figure 3. Initially, the FFT algorithm is performed on the luminance part of the image. Then, the coefficients of the Fourier magnitude phase are extracted along the radius r positions, and the maximum value of normalized cross-correlation $Cmax$ is estimated between the extracted coefficients F and the sequence W of the watermark:

$$C_{max} = Max_{0 \le j \le N-1} \left(\frac{\sum_{i=0}^{N-1} \left(W(i) - W \right) \left(F(i+j) - F \right)}{\sqrt{\sum_{i=0}^{N-1} \left(W(i) - W \right)^2 \sum_{i=0}^{N-1} \left(F(i+j) - F \right)^2}} \right) \tag{5}$$

where N is the sequence length, W and F are, respectively, the means of the watermark and extracted Fourier coefficients in the magnitude phase. The watermark is detected if the maximum value of the normalized cross-correlation exceeds a predefined threshold t.

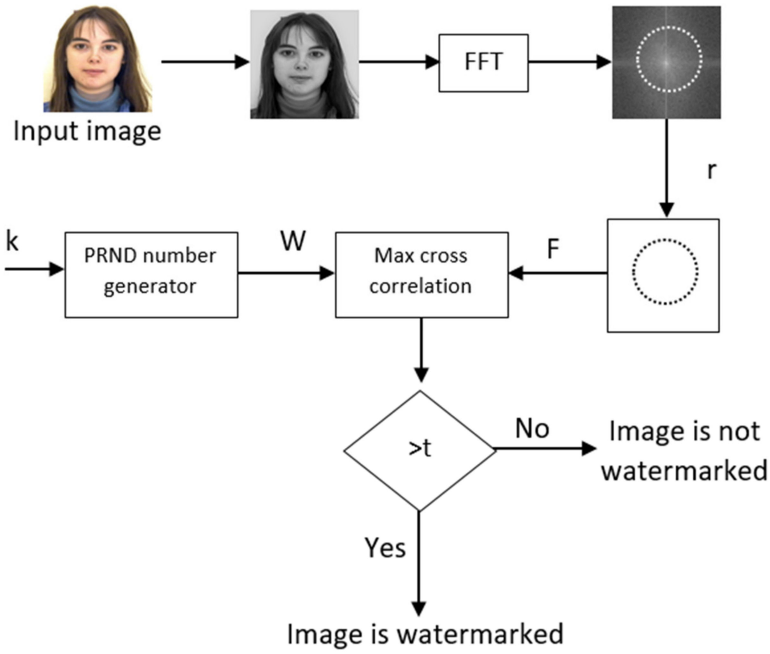


Figure 3. Fourier watermark detection scheme.

4. Print-Cam Watermarking Application

This section, divided by subheadings, provides a concise and precise description of the proposed application, as well as the experimental results and conclusions that can be drawn.

4.1. Projective Correction Pre-Process

The action of taking a picture with a smartphone creates a geometric relation between the object and the camera position in the capturing process. This relation is called projective transform, where the point's coordinates from a 3D world are mapped into a 2D image plan [33]. That is, 2D projective transformation, or perspective transformation, is the projection from 2D plan in the 3D world into the 2D plan (image plan), and this is exactly our case study since the capturing process is identified as taking a photo of a physical printed image, which is a 2D object. This relation is indicated as follows:

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = H \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}, \tag{6}$$

H defines the 3×3 perspective matrix. This matrix describes the parameters of the geometric distortion in a print-cam process. Therefore, from the mathematical perspective, to delete or invert the geometric attacks, those parameters should be estimated. The projective matrix has eight degrees of freedom, and hence eight unknown parameters should be estimated. The system equation that we need to solve is:

$$\begin{cases} x = \frac{h_{11}X+h_{12}Y+h_{13}}{h_{31}X+h_{32}Y+1}, \\ y = \frac{h_{21}X+h_{22}Y+h_{23}}{h_{31}X+h_{32}Y+1}. \end{cases} \tag{7}$$

As a starting step, Hough transformation is applied to detect lines of the frame in the captured image. Then, the four corners are computed using the intersections of those lines. The projective matrix is found by solving the system of Equation (7) by applying the four estimated points. Finally, the geometric inverse transformation is performed on the image to reconstruct the corrected image. Figure 4 shows an example of the projective rectification.

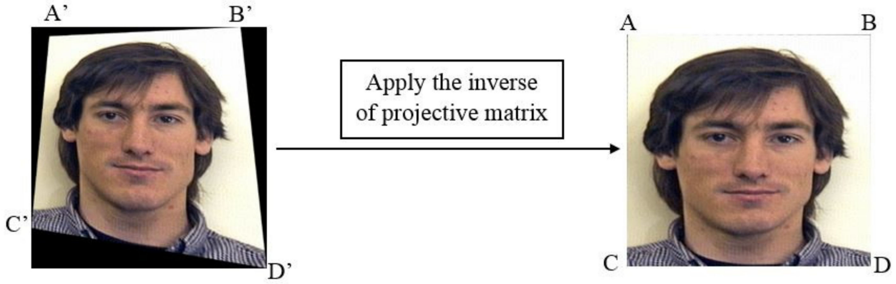


Figure 4. Projective rectification.

To delete the geometric effect on the captured image, the four corner positions need to be found. Therefore, the following steps of this process are completed:

Step 1. Detect the four corners: we use Hough line to detect the frame of the ID image, and then we get the four points from the intersections of those lines.
Step 2. Estimate the projective matrix: with the corresponding four points, we solve the system of Equation (7).
Step 3. Apply the inverse transformation in the whole image to reconstruct the image with no geometric effect.

4.2. System Design and Implementation

In this section, we present a design and implementation of the proposed watermarking method application. The application contains two main operations: one for the insertion phase and the other for detection phase. Since the system we are dealing with is a print-cam system, the geometric pre-process rectification is integrated in the detection phase.

The implementation is an Android application to run on Android-based smartphones. The application is implemented using the Java programming language with Android SDK. All are tested and executed on emulators, as well as real Android devices. The conceptual flow-work of our watermarking technique is shown in Figures 5 and 6.

The watermark insertion process is composed of the following major functions (Figure 5):

1. Importing the original host image: the user chooses an image existing already in the gallery of the smartphone to apply image watermarking on it.
2. Watermark insertion: the watermark is generated in this phase by default-predefined parameters, and the luminance part of the chosen image is processed as shown in Figure 2.
3. Saving the watermarked image: the watermarked image is converted back to its original RGB color type, exported, and saved on the user's smartphone, ready for the printing step

At this step, the watermark is detected after printing the image on a physical support. The watermark detection process is performed as follows (Figure 6):

1. Taking a picture: at this stage, the user takes a picture freehandedly of the printed watermarked image with the integrated camera of the smartphone.
2. The captured image is named and saved in the application folder and ready for watermark detection.
3. The detection phase combines projective correction of the captured image and watermark detection, as shown in Figure 3.

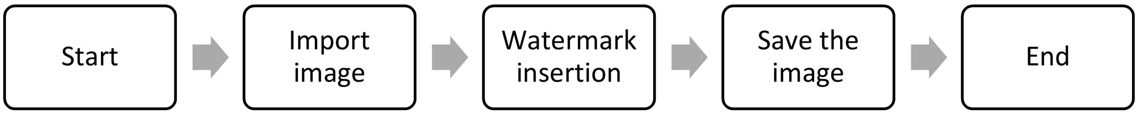


Figure 5. Flow process of watermark insertion in the proposed application.

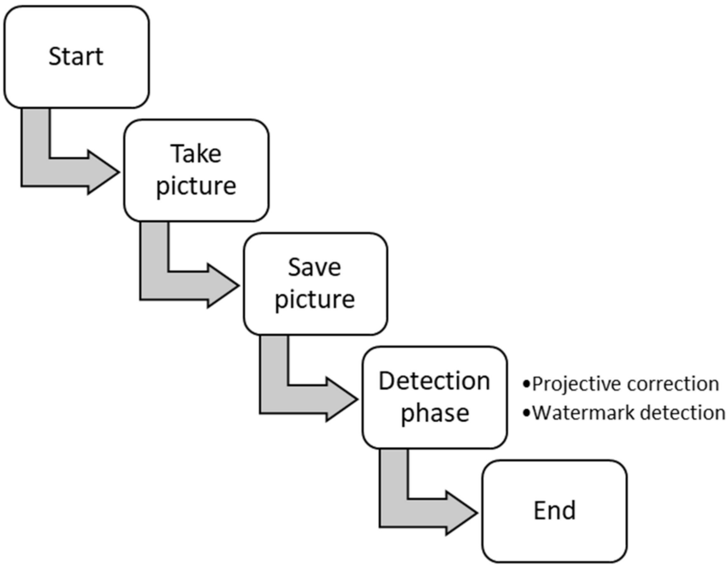


Figure 6. Flow process of watermark detection in the proposed application.

4.3. Testing and Performance Analysis

The proposed Android application was implemented using Android Studio IDE, Java programming language, and OpenCV, an open-source image-processing library.

4.3.1. Application User Interface

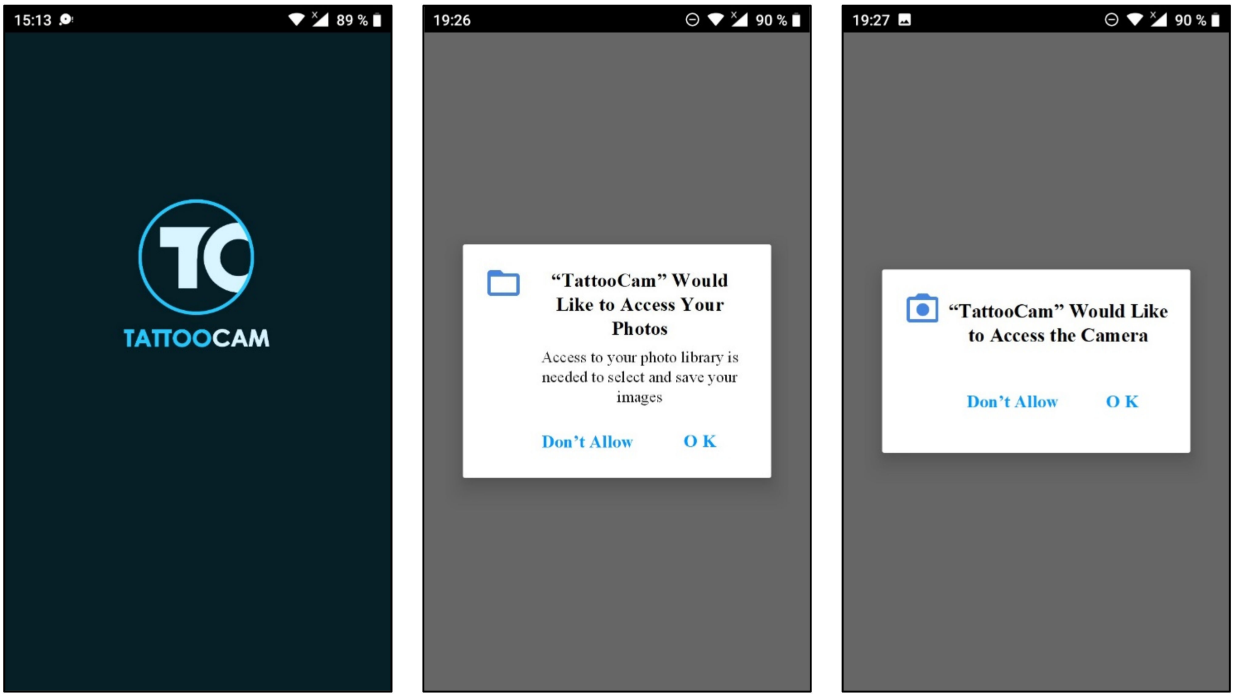
The main scenarios of the developed application are shown below:

1. We named the application "TattooCam." While downloading the application, ask for authorization to use the gallery and the camera of the device, as shown in Figure 7.
2. Watermark insertion scenario: the main activities of watermark insertion process designed in the application are shown in Figure 8.
3. Watermark detection scenario: the main activities of watermark detection process designed in the application are shown in Figure 9.

4.3.2. Application Performance

The image database used to test the application is from the PICS images (Psychological Image Collection at Stirling-Aberdeen) [34]. The images have a resolution of between 336×480 and 624×544, and they are scaled to 512×512, which tolerates the fast Fourier transform (FFT) algorithm (Figure 10). The parameters of the Fourier watermarking method are predetermined as follows: 40 dB for PSNR, the radius r of the circular position, where the watermark can be embedded, is equal to 70 and the length of the watermark is equal to 180.

The images are printed on a semi-rigid 300 g paper with a Konica Minolta bizhub C754e printer with a resolution of 200 dpi. The printed ID images are 44×44 mm^2. The printed images are captured freehandedly with a Nokia TA-1020 smartphone having a camera with a resolution of 8 megapixels. The smartphone's camera has default settings and the pictures are captured in daylight luminance. The characteristics of tools used for the development of the application are shown next in Tables 1 and 2.



(**a**) (**b**) (**c**)

Figure 7. TattooCam application: (**a**) start the application; (**b**) access authorization to use the images of the device; (**c**) access authorization to use the camera of the device.

Table 1. Hardware specifications of the computer used for developing the application.

Computer	Specifications
Brand	HPEliteBook 8440 p
Processor	Intel Core i5 2.40 GHz
RAM	6 Go
Storage	230 Go
Operating system	Windows 10 Professionnel

Table 2. Hardware specifications of the tested smartphone.

Smartphone	Specifications
Brand	Nokia TA-1020
Processor	Quad-core 1.25 GHz Cortex-A53
RAM	2 Go
Storage	16 Go
Operating system	Android 9

The following tables (Tables 3 and 4) show the correlation values of watermarked images and non-watermarked images, along with the execution of the TattooCam application before and after applying the print-cam process.

Table 3. The correlation values of images before and after the insertion process.

Images	Maximum Correlation Values	
	Without Watermark	With Watermark
Alison	0.2571	0.9089
Andrew	0.3133	0.8840
Caroline	0.3517	0.8802
Barry	0.2296	0.9007

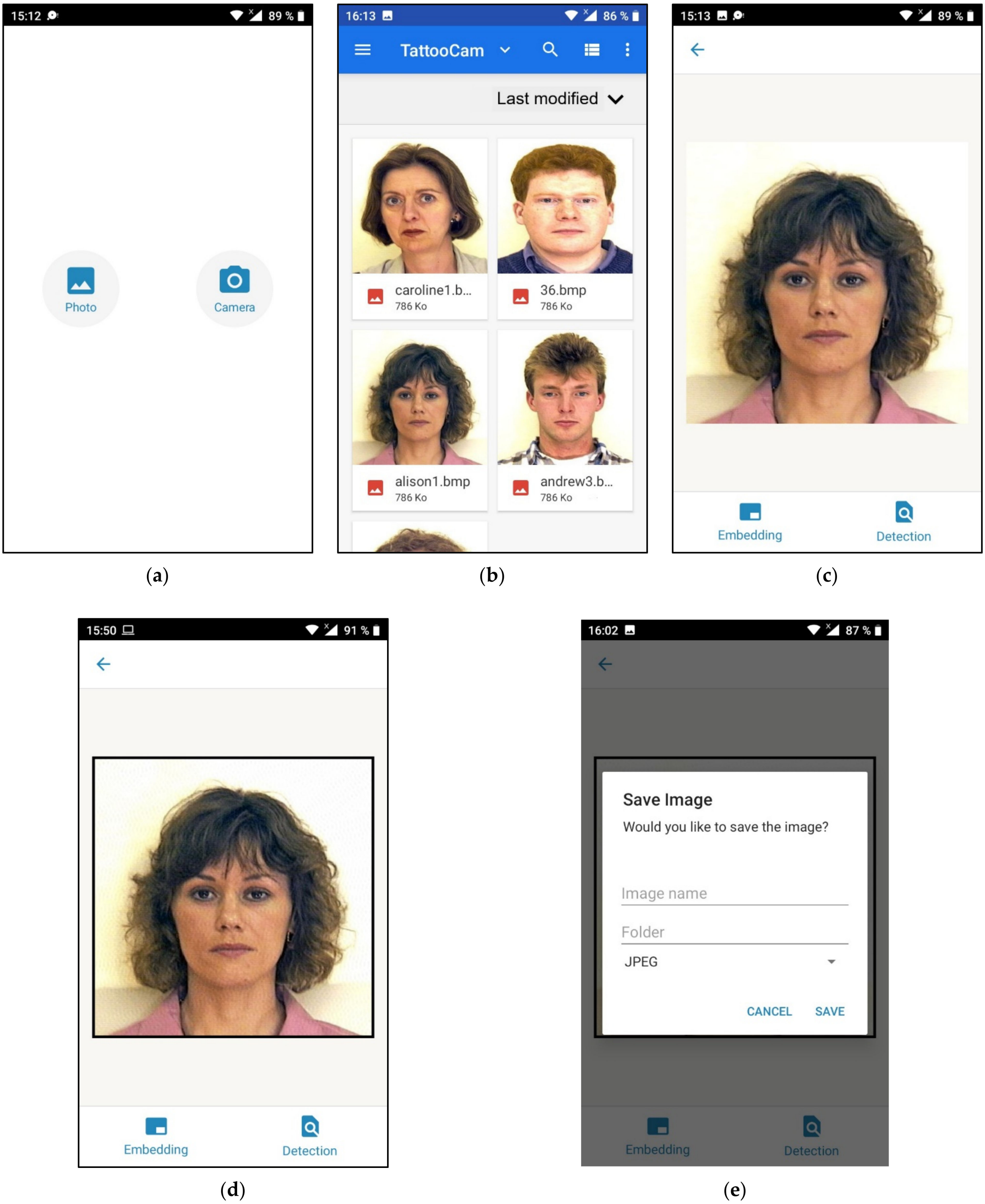


Figure 8. Watermark insertion process: (**a**) the main activity where the user can press the photo to access the gallery; (**b**) folder with the searched images; (**c**) the chosen image is displayed on other activity for watermark insertion; (**d**) after pressing the embedding key, the watermarked image is displayed on the screen with a frame; (**e**) action to name and save the new watermarked image.

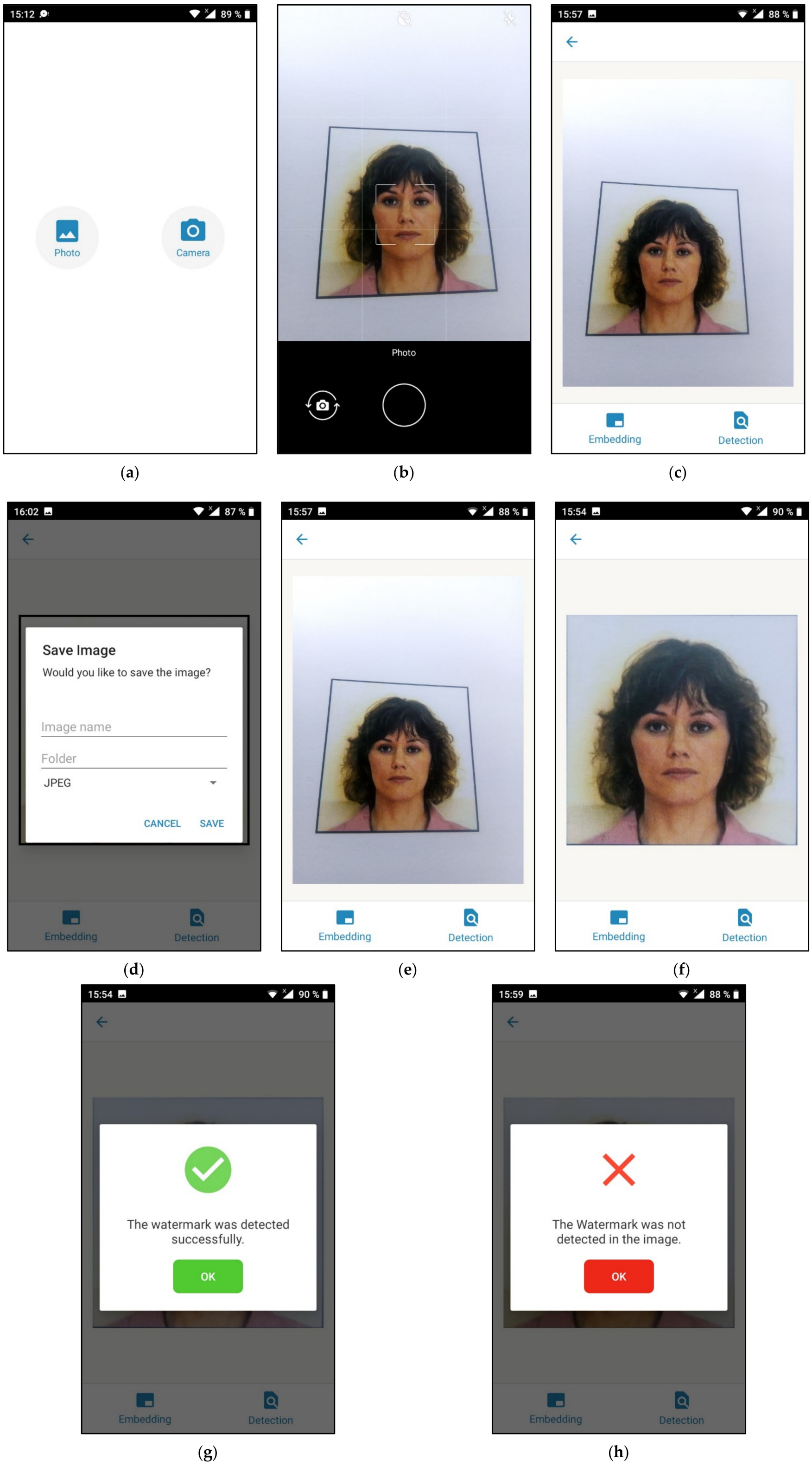


Figure 9. Watermark detection process: (**a**) the main activity where the user can press the camera key to access the camera of the device; (**b**) user takes picture freehandedly; (**c**) the captured image is displayed in the main screen of the application; (**d**) activity to name and save the captured image; (**e**) the captured image is displayed and the user can press detection; (**f**) geometrically corrected image is displayed; (**g**) detection decision of the existence of the watermark in the captured image; (**h**) detection decision of the absence of the watermark in the image.

Figure 10. Set of tested images.

Table 4. The correlation values of images after the print-cam process.

Images	Maximum Correlation Values	
	Without Watermark	**With Watermark**
Alison	0.1794	0.1794
Andrew	0.2456	0.5112
Caroline	0.2253	0.5523
Barry	0.1974	0.5148

The above table shows the robustness of the proposed application against projective distortions since the images are captured freehandedly and existing attacks in the device, such as JEPG compression blur and color degradations, have no prior modification from the user. The average value of normalized cross-correlation values exceeds 0.5, which is more than the average values in the pre-existing print-cam watermarking methods [3].

In Table 5, the results show the execution time of the proposed watermarking application for the tested set of images.

Table 5. Time consumption of the TattooCam application.

Images	Time Consumption (Second)	
	Without Watermark	**With Watermark**
Alison	10.07 s	9.5 s
Andrew	4.10 s	7.44 s
Caroline	6.54 s	5.84 s
Barry	4.11 s	6.45 s

As shown in the table above, the total elapsed time for the combined phases of the watermarking application (insertion and detection) does not exceed 10 s with an average of 7 s. However, the average time consumption of about 21 s is considered acceptable for performing the whole watermarking process in an industrial field [23].

5. Conclusions

In this paper, we have proposed an application of a blind and imperceptible image watermarking based on Fourier transform for print-cam systems. This system presents a process where watermarked ID images are printed and carried out on physical medium and then captured freehandedly with a smartphone camera for watermark detection. The application is implemented as an Android application for smartphones with the name "TattooCam". The method contains a pre-process stage to correct the projective deformations of images captured freehandedly. For an industrial application concerning the security of ID images, the application shows robustness against print-cam attacks, including projective deformations, JEPG compression, blur, and color degradation, as well as an acceptable average time consumption during the execution of different phases in the application. Further, the proposed application does not require prior knowledge of the materials (printer, smartphone, and camera resolution). All the above fulfill the use of the developed method in real-time industrial applications. Based on these results, the next step will focus on the improvement of the application to deal with other, more complex print-cam attacks, such as taking pictures under luminance variation, as well as improvement of the other watermarking requirements, such as security and capacity.

Author Contributions: Conceptualization, K.G.; methodology, K.G., F.R., H.D., R.H. and R.R; software, K.G., F.R. and R.R.; validation, K.G., F.R., H.D., R.H. and R.R.; formal analysis, K.G.; investigation, K.G. and R.R.; resources, K.G. and R.R.; data curation, K.G. and R.R.; writing—original draft preparation, K.G.; writing—review and editing, K.G., F.R., H.D., R.H. and R.R.; visualization, K.G. and R.R.; supervision, F.R., H.D., R.H. and R.R; project administration, H.D. and R.H.; funding acquisition, R.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data is available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cox, I.; Miller, M.; Bloom, J.; Fridrich, J.; Kalker, T. *Digital Watermarking and Steganography*; Morgan Kaufmann: Burlington, MA, USA, 2007.
2. Muhle, F. *Christian Fuchs: Social Media. A Critical Introduction*; M&K Medien & Kommunikationswissenschaft; Sage Publications: Los Angeles, CA, USA, 2014; Volume 62, pp. 662–663.
3. Gourrame, K.; Douzi, H.; Harba, R.; Riad, R.; Ros, F.; Amar, M.; Elhajji, M. A zero-bit Fourier image watermarking for print-cam process. *Multimed. Tools Appl.* **2019**, *78*, 2621–2638. [CrossRef]
4. Pramila, A.; Keskinarkaus, A.; Seppänen, T. Watermark robustness in the print-cam process. In *Proceedings of the IASTED Signal Processing, Pattern Recognition, and Applications (SPPRA 2008)*; Citeseer: Anaheim, CA, USA, 2008; pp. 60–65.
5. Begum, M.; Uddin, M.S. Analysis of digital image watermarking techniques through hybrid methods. *Adv. Multimed.* **2020**, *2020*, 7912690. [CrossRef]
6. Riad, R.; Harba, R.; Douzi, H.; Ros, F.; Elhajji, M. Robust fourier watermarking for id images on smart card plastic supports. *Adv. Electr. Comput. Eng.* **2016**, *16*, 23–30. [CrossRef]
7. Chen, W.; Zhu, C.; Ren, N.; Seppänen, T.; Keskinarkaus, A. Screen-Cam Robust and Blind Watermarking for Tile Satellite Images. *IEEE Access.* **2020**, *8*, 125274–125294. [CrossRef]
8. Poljicak, A.; Mandic, L.; Agic, D. Discrete Fourier transform-based watermarking method with an optimal implementation radius. *J. Electron. Imaging* **2011**, *20*, 33008. [CrossRef]
9. Bas, P.; Chassery, J.M.; Macq, B. Geometrically invariant watermarking using feature points. *IEEE Trans. Image Process.* **2002**, *11*, 1014–1028. [CrossRef] [PubMed]
10. Fang, H.; Chen, D.; Huang, Q.; Zhang, J.; Ma, Z.; Zhang, W.; Yu, N. Deep Template-based Watermarking. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *31*, 1436–1451. [CrossRef]
11. Lee, M.J.; Kim, K.S.; Oh, T.W.; Lee, H.Y.; Lee, H.K. Improved watermark synchronization based on local autocorrelation function. *J. Electron. Imaging* **2009**, *18*, 23008. [CrossRef]
12. Lichtenauer, J.F.; Setyawan, I.; Kalker, T.; Lagendijk, R.L. Exhaustive geometrical search and the false positive watermark detection probability. InSecurity and Watermarking of Multimedia Contents V. *Int. Soc. Opt. Photonics* **2003**, *5020*, 203–214.
13. Nakamura, T.; Katayama, A.; Yamamuro, M.; Sonehara, N. Fast watermark detection scheme for camera-equipped cellular phone. In Proceedings of the 3rd International Conference on Mobile and Ubiquitous Multimedia, New York, NY, USA, 27 October 2004; pp. 101–108.
14. Katayama, A.; Nakamura, T.; Yamamuro, M.; Sonehara, N. New high-speed frame detection method: Side trace algorithm (STA) for i-appli on cellular phones to detect watermarks. In Proceedings of the 3rd International Conference on Mobile and Ubiquitous Multimedia, New York, NY, USA, 27 October 2004; pp. 109–116.
15. Takeuchi, S.; Kunisa, A.; Tsujita, K.; Inoue, Y. Geometric distortion compensation of printed images containing imperceptible watermarks. In *2005 Digest of Technical Papers, Proceedings of the International Conference on Consumer Electronics, Las Vegas, NV, USA, 12 January 2005*; ICCE: Las Vegas, NV, USA, 2005; pp. 411–412.
16. Pramila, A.; Keskinarkaus, A.; Seppänen, T. Toward an interactive poster using digital watermarking and a mobile phone camera. *Signal. Image Video Process.* **2012**, *6*, 211–222. [CrossRef]
17. Pramila, A.; Keskinarkaus, A.; Takala, V.; Seppänen, T. Extracting watermarks from printouts captured with wide angles using computational photography. *Multimed. Tools Appl.* **2017**, *76*, 16063–16084. [CrossRef]
18. Thongkor, K.; Amornraksa, T. Digital image watermarking for photo authentication in Thai national ID card. In Proceedings of the 2012 9th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, Phetchaburi, Thailand, 16 May 2012; pp. 1–4.

19. Thongkor, K.; Amornraksa, T. Robust image watermarking for camera-captured image using image registration technique. In Proceedings of the 2014 14th International Symposium on Communications and Information Technologies (ISCIT), Incheon, Korea, 24 September 2014; pp. 479–483.
20. Adjabi, I.; Ouahabi, A.; Benzaoui, A.; Taleb-Ahmed. Past, present, and future of face recognition: A review. *Electronics* **2020**, *9*, 1188. [CrossRef]
21. Adjabi, I.; Ouahabi, A.; Benzaoui, A.; Jacques, S. Multi-Block Color-Binarized Statistical Images for Single-Sample Face Recognition. *Sensors* **2021**, *21*, 728. [CrossRef] [PubMed]
22. Isa, M.R.M.; Aljareh, S.; Yusoff, Z. A watermarking technique to improve the security level in face recognition systems. *Multimed. Tools Appl.* **2017**, *76*, 23805–23833. [CrossRef]
23. Al-Otum, H.; Al-Shalabi, N.E. Copyright protection of color images for android-based smartphones using watermarking with quick-response code. *Multimed. Tools Appl.* **2018**, *77*, 15625–15655. [CrossRef]
24. Takeshita, S.; Maehara, T.; Ono, S. Digital watermark design for two-dimensional codes displayed on smart phone screen using multi-objective optimization and optical simulation. In *International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*; Springer: Cham, Switzerland, 2017; pp. 201–214.
25. Le, N.T. Invisible watermarking optical camera communication and compatibility issues of IEEE 802.15. 7r1 specification. *Opt. Commun.* **2017**, *390*, 144–155. [CrossRef]
26. Pramila, A.; Keskinarkaus, A.; Seppänen, T. Increasing the capturing angle in print-cam robust watermarking. *J. Syst. Softw.* **2018**, *135*, 205–215. [CrossRef]
27. Digimarc. Discover Application. 2015. Available online: https://www.digimarc.com/products/discover/mobile-app (accessed on 17 October 2021).
28. Pereira, S.; Pun, T. Robust template matching for affine resistant image watermarks. *IEEE Trans. Image Process.* **2000**, *9*, 1123–1129. [CrossRef]
29. Delgado-Guillen, L.A.; Garcia-Hernandez, J.J.; Torres-Huitzil, C. Digital watermarking of color images utilizing mobile platforms. In Proceedings of the 2013 IEEE 56th International Midwest Symposium on Circuits and Systems (MWSCAS), Columbus, OH, USA, 4 August 2013; pp. 1363–1366.
30. Nguyen, P.B.; Beghdadi, A.; Luong, M. Robust watermarking in DOG scale space using a multi-scale JND model. In *Pacific-Rim Conference on Multimedia*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 561–573.
31. Chen, W.; Ren, N.; Zhu, C.; Keskinarkaus, A.; Seppänen, T.; Zhou, Q. Joint Image Encryption and Screen-Cam Robust Two Watermarking Scheme. *Sensors* **2021**, *21*, 701. [CrossRef] [PubMed]
32. Riad, R.; Ros, F.; Harba, R.; Douzi, H.; El Hajji, M. Pre-processing the cover image before embedding improves the watermark detection rate. In Proceedings of the 2014 Second World Conference on Complex Systems (WCCS), Agadir, Morocco, 10 November 2014; pp. 705–709.
33. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer, Vision*, 2nd ed.; Cambridge University Press: New York, NY, USA; Cambridge, UK, 2003.
34. Hancock, P. Psychological Image Collection at Sirling (pics). 2008. Available online: http://pics.psych.stir.ac.uk (accessed on 20 April 2021).