High Dynamic Range Image Sharing with Privacy Protection

Evgeniy Upenik, Pol Fernandez-Marti, and Touradj Ebrahimi

Multimedia Signal Processing Group (MMSPG)
Ecole Polytechnique Fédérale de Lausanne (EPFL)
CH-1015 Lausanne, Switzerland
Email: firstname.lastname@epfl.ch

ABSTRACT

Sharing pictures has become a very popular practice among consumers. Most recent cameras, displays, and smartphones can capture and display images in high dynamic range and wide colour gamut, contributing to an increase of this type of content. It is a well-known fact that pictures contain information that could cause various security and privacy issues. This is in particular true because high dynamic range pictures exhibit more information when compared to conventional images resulting in an increased invasion of privacy. Lack of proper solutions to overcome privacy can hinder a wide spread adoption of high dynamic range pictures shared in social networks. For example, dark areas of a content that are otherwise difficult to view in a conventional picture become more visible. It is therefore natural that mechanisms are offered to protect privacy while sharing pictures also in high dynamic range. In this paper, we propose an architecture to share high dynamic range images captured by high-end smartphones while offering a mechanism to protect privacy. In our approach, the high dynamic range picture is tone-mapped into a lower dynamic range version which reduces its degree of invasiveness as far as privacy is concerned. Eventually, any remaining privacy sensitive areas in the picture can be further protected (blurring of faces, masking cars licence plates, etc.). The core of the proposed architecture conforms to both JPEG XT, a recently standardised format that offers backward compatibility with legacy JPEG and to JPEG Systems under development. The proposed solution allows to publicly share a protected version of the high dynamic range image tone-mapped to a standard dynamic range picture. The latter can go through transmorphing operation to further protect it. Authorised users can access the original high dynamic range picture. The architecture as described above has been implemented as an app for smartphones running Android OS. We demonstrate feasibility and usefulness of this approach and discuss its advantages when compared to conventional image sharing such as those used in social networks.

Keywords: image processing, privacy, HDR, JPEG XT, JPEG Systems, security, social networks

1. INTRODUCTION

Social networks are a big part of the everyday life for almost every person today. And the wide spread of smart mobile devices with high-resolution cameras and user-friendly social networks make photo and video sharing an easy and popular activity. At the same time, however, it raises a wide concern on privacy issues since this shared content potentially reveals a great amount of sensitive information about individuals.

High Dynamic Range (HDR) imaging, as opposed to Standard Dynamic Range (SDR), which is one of the trends in the evolution of visual content representation technologies, has already become reality in consumer devices thanks to advancements in both capture and display technologies. Pictures that are captured in HDR contain more information: for instance, the very dark or very bright areas which would not be visible if stored in SDR can contain more visual information that is potentially sensitive. This increases the level of possible privacy concerns.

HDR images need to be stored differently than their SDR counterparts. Thus, other formats than the legacy JPEG have been developed and standardized. JPEG XT is a still-image coding technology that allows the storage

---

Send correspondence to E.U. and T.E.
P.F-M. contributed to this paper while being an exchange student at EPFL

and distribution of true high dynamic range data efficiently and in a compact format. It is backward compatible with legacy JPEG, which makes it an attractive solution, and has been standardized as ISO/IEC 18477.

A framework that addresses the privacy issues related to the use of image sharing in social networks has been proposed by Yuan et al.[1,2] In particular, its implementation in ProShare framework allows users to share photos while maintaining their privacy. However, this framework is not able to support HDR images and there is no compatibility with the JPEG XT. Thus, the goal of this paper is to integrate HDR imaging support using the JPEG XT standard into the framework of the ProShare implementation, and to provide the ability of secure image sharing with HDR quality.

The remainder of this paper is organized as follows: Section 2 introduces the basic concepts, including details on the way a human eye perceives the world and how this may be represented and displayed with HDR imaging. Section 3 explains the current technology used in ProShare framework to protect and share images. Section 4 explains the necessary modifications that were applied in the ProShare framework in order to integrate HDR support using JPEG XT standard. And finally, Section 5 concludes the paper and discusses possible future work.

2. THEORETICAL AND TECHNOLOGICAL BACKGROUND

2.1 Human Visual System

In order to see, the human eye is composed of multiple parts. The cornea of our eye acts as a lens and refracts the light, and the iris contracts and expands in order to adapt to the scene's luminance, just as a camera aperture. The pupil allows to adjust captured light intensity by varying its aperture diameter from 2 to 8 mm. The lens focuses on a targeted subject by changing its shape and thus changing the focal distance of the eye. The retina receives rays of light onto its photoreceptors (rods and cones), which in turn convert the light into neurological signals.

Rods are very sensitive to light and are spread out unevenly on the retina except its center. Rods are well suited for low light conditions, motion, and peripheral vision. However, they do not assist in capturing color information. The dynamic range over which the rods can operate spans between 10^{-6} and 10 cd/m^2. Cones are optimal for vision in brighter light and allow humans to distinguish between ten million different colors. The cones can be stimulated by light of luminance levels ranging from approximately 0.03 to 10^8 cd/m^2.[3]
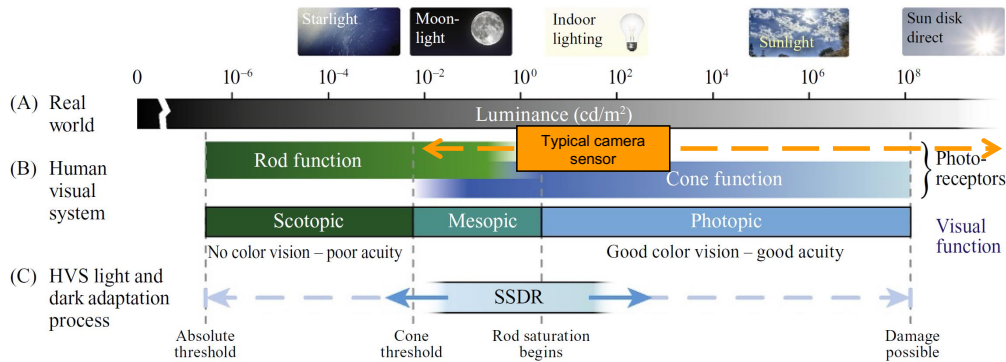

Figure 1. Dynamic range of Human Visual System (HVS) and real world[3] compared to a typical camera sensor (orange).

The human eye is able perceive a greater dynamic range than a typical photo camera (Figure 1). If we were to consider HVS light and dark adaptation to adjust to a scene's luminance, our eyes can see over a range of nearly 24 f-stops. This would not be, however, a fair comparison for digital cameras. If human eyes were to stare at a single point of a physical scene, estimations suggest that it would have a dynamic range between 10 and 14 f-stops, meaning that between 1000 and 16000 levels of tonal precision would be necessary in order to accurately represent the dynamic range of a scene.

2.2 High Dynamic Range Imaging

The dynamic range of a scene in photography is the ratio of the highest scene luminance to the lowest scene luminance. The high dynamic range (HDR) is broader than what is considered to be the standard dynamic range (SDR), also called as low dynamic range (LDR). An SDR image is considered as an image that can be captured with a regular camera and is usually encoded in 8-bit per channel components. HDR imaging aims to capture a wider dynamic range of luminance values to better convey the real-world user experience. This information cannot be always stored in 8-bit per channel components and, thus, often a higher bit-depth is required.

2.2.1 HDR image acquisition

Most cameras and smartphones cannot capture an HDR image in a single shot, due to limitations related to the dynamic range of their sensor. For this reason HDR images are typically acquired by taking several shots of the same scene with varying exposure parameters. These shots are fused in order to reconstruct a high dynamic range image of the luminance levels of the real world. Tone mapping is a compression of the dynamic range that is used to approximate the appearance of HDR images in a medium that has a more limited dynamic range.

2.2.2 Displaying HDR

To be able to display an HDR image in its full potential, a display with a high contrast ratio is needed. HDR for television broadcasting has emerged in the last years, and the main manufacturers adopted rather rapidly this technology in their newer models. Smartphone manufacturers also adopted the HDR technology to improve the viewing experience of their users.

The two main technologies used in smartphone displays are Liquid Crystal Display (LCD) and Organic Light-Emitting Diode (OLED). LCD panels use a backlight to illuminate the pixels, and can provide very high levels of brightness. In contrast to LCD panels, OLED displays are always off unless the individual pixels are electrified. OLED displays can provide darker blacks, because if an OLED pixel is not receiving electricity, it does not emit any light and, therefore, remains more black. The OLED displays, typically, cannot achieve higher brightness levels than the LCD displays. Although both technologies can display HDR content, the majority of the HDR devices (including high-end smartphones) currently use OLED displays.

There exist several formats to display HDR and are widely adopted by mobile mobile phones and displays manufacturers.

- **HDR10**[4] is the most widely used HDR format, which increases the color depth of media formats from the 8-bit used in standard dynamic range to 10-bit RGB for HDR content. It also supports up to 4000 nits peak brightness. It uses static metadata for video content. Moreover, it is an open source format.

- **Dolby Vision**[5] uses 12 bits and can go up to 10'000 nits of peak brightness. The fundamental difference between the Dolby Vision and the HDR10 formats in video content is that the former allows the addition of dynamic metadata and adaptation of the HDR image on a frame-by-frame basis. This information is associated with the content that is reproduced and indicates precisely to the device how the brightness should be treated.

- **HDR10+**[6] is an extension of HDR10 that uses 10-bit color depth, and supports up to 10'000 nits peak brightness. The difference with HDR10 is the use of dynamic metadata for video content. Initially, it was supported by Samsung and Amazon, but a growing number of companies are adopting it because, similarly to HDR10, it is an open source standard.

- **Hybrid Log Gamma** (HLG) was created by BBC and NHK to broadcast HDR content. It is backwards-compatible with SDR devices. HLG broadcasts a standard dynamic range (SDR) signal that any TV can use. Extra information is added for HDR rendering in order to be used by HDR devices. HLG uses the same gamma curve as an SDR signal but adds a logarithmic curve with extra brightness on top of the signal.[7]

2.3 JPEG XT Standard

JPEG XT (XT is short for eXTension) is an image compression standard which specifies backward-compatible extensions of the legacy JPEG standard.[8]

These extensions include HDR, compression for high bit-depth content, lossless and near-lossless coding as well as alpha channel coding. JPEG XT uses application markers (APP markers) of the legacy JPEG structure to embed additional information. The original intent of these markers was to let applications add vendor-specific information to the bitstream without breaking compatibility. An example of a commonly used APP marker is APP1 with EXIF metadata. Decoders unaware of a specific application marker would ignore it and skip over it. In the case of JPEG XT, these markers are used to hide JPEG XT related information from legacy JPEG decoders. The main principle of the concept is to cut a box structure into parts smaller than 64 KB as this is the limitation of an APP marker segment and to include these parts into multiple instances of APP11 marker segments.

2.3.1 JPEG XT for HDR

The legacy JPEG specifications include a 12-bit mode and the lossless JPEG coding mode supports up to 16 bits per sample. Unfortunately, these two JPEG modes are incompatible with the popular 8-bit mode and are not used in popular applications, such as digital photography. HDR images require more than 8 bits per sample, and those SDR JPEG images are proving to be a bottleneck in the still-imaging ecosystem, as monitors, smartphones, and other portable devices are now capable of supporting HDR. Smartphones offer a so-called HDR mode, taking multiple exposures and combining them using proprietary tone-mapping to improve color and contrast. However, the final image is still 8-bit JPEG. The true HDR information is discarded.

JPEG XT allows storage of more than 8 bits per component images and can be used to store HDR images. JPEG XT defines 4 profiles to encode HDR images. Profiles A, B and C use 2 layers (illustrated in Figure 2). The base layer stores a traditional tone-mapped 8-bit JPEG image, and the extension layer (also called residual layer) includes the additional information to reconstruct the HDR image, this information is also encoded as a legacy 8-bit image and included in the APP11 marker. As a result, legacy JPEG implementations must be able to decode an SDR image from a JPEG XT bitstream.
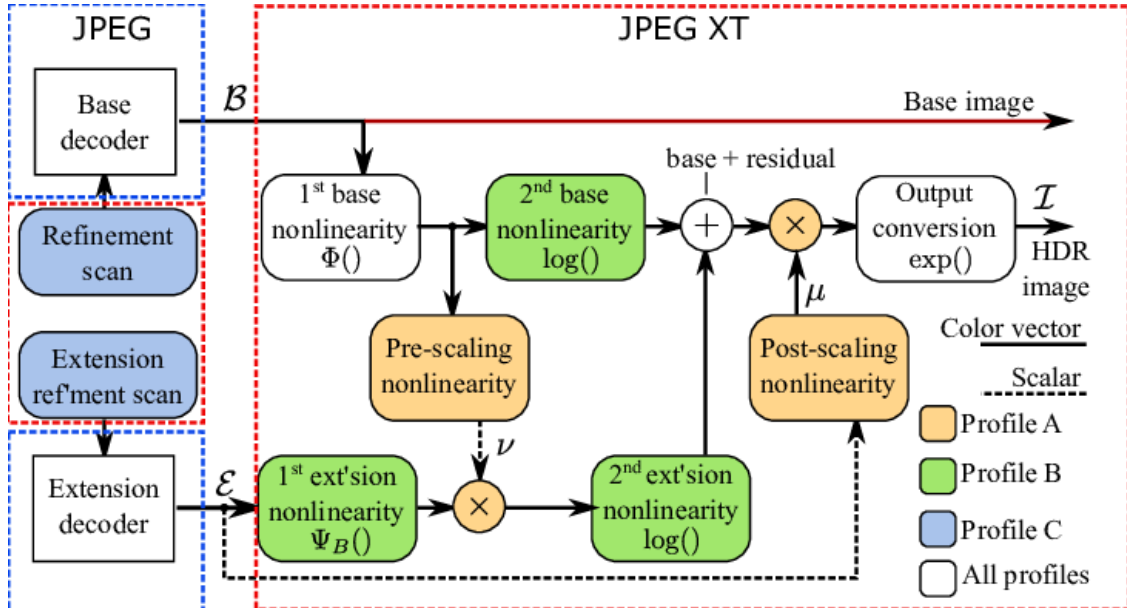

Figure 2. The simplified architecture of Profiles A, B and C. (blue-dashed line) is the legacy JPEG decoder and (red-dashed line) are the additional components that define the new JPEG XT standard.[8]

Profile D is different from the three previous profiles and it does not include two separate images. Profile D extends the resolution of the image precision by extending the precision of the DCT components of the legacy JPEG. The simplified process of Profile D decoding is illustrated in Figure 3.
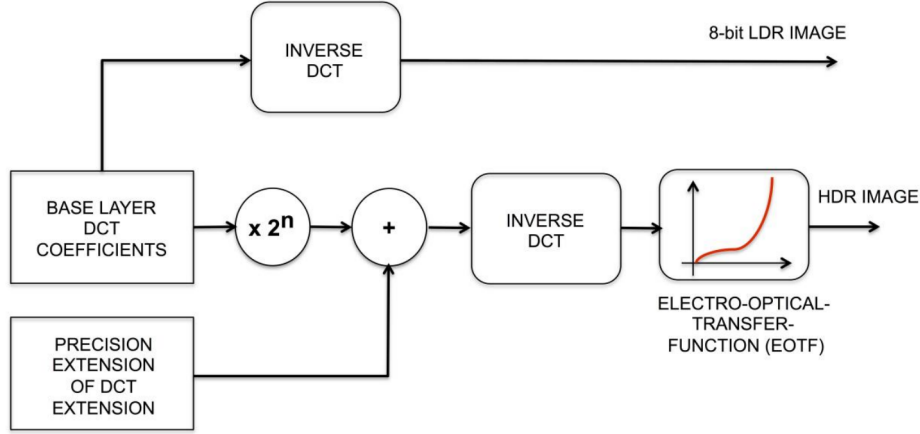

Figure 3. The simplified architecture of Profile D.[8]

3. TRANSMORPHING FOR PRIVACY PROTECTION

Secure JPEG transmorphing is a method that was proposed by Yuan et al. in 2015.[1,2] Transmorphing uses the JPEG application segments markers (APPn) in file header to secretly store partial original image information, while encoding the JPEG image in a visually protected form. The visual information of the original image can be protected by any type of regional manipulation, such as masking, blurring, pixelation, inpainting, warping, and so on. The protected image, or as called the transmorphed image, is backward compatible with legacy JPEG. With a dedicated JPEG transcoder or decoder that supports JPEG transmorphing, the original image can be recovered by replacing the obfuscated regions in the protected image with the corresponding original regions extracted from APPn markers.
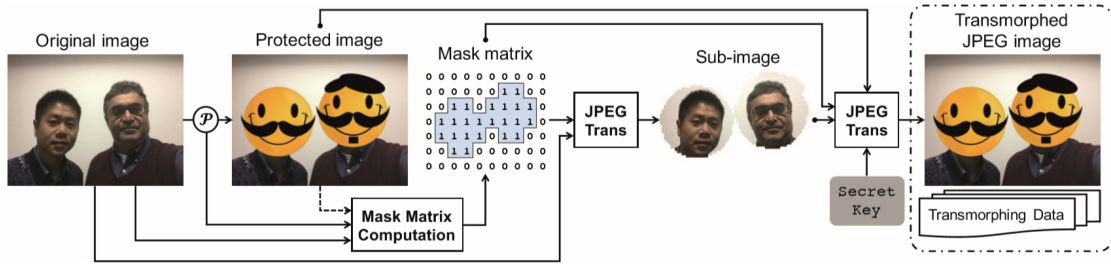

Figure 4. JPEG Transmorphing algorithm[2]

3.1 Transmorphing Method

The architecture of the JPEG transmorphing algorithm is illustrated in Figure 4. The whole algorithm consists of two processes: transmorphing and reconstruction (also called re-transmorphing). The transmorphing protection procedure of secure JPEG transmorphing consists of three steps: (i) mask matrix generation, (ii) sub-image construction and (iii) transmorphing data insertion.

Mask matrix generation: Firstly, user obfuscates certain regions of interest (ROIs) of a given image using arbitrary regional manipulation, such as masking face regions with cartoon stickers as example images in Figure 4. Once the obfuscation has been applied, a binary-valued two dimensional matrix is generated, which

specifies the shape, size and position of protected ROIs. Each element of the matrix corresponds to a Minimum Coded Unit (MCU) block of the upcoming encoded JPEG image, where elements 1 indicate protected blocks and 0 unprotected. This matrix is called Mask Matrix, which holds essential geometrical information about the image ROIs being protected. Depending on applications, the mask matrix can be generated by comparing the original and obfuscated images.

Sub-image construction: Secondly, a sub-image is constructed by encoding or transcoding the original image to a new JPEG image, during which DCT coefficients corresponding to the ROIs defined by the Mask Matrix are preserved while other coefficients outside the ROIs are set to zero. The sub-image is still a JPEG image with the same dimensions but smaller file size when compared to the original image. It contains information about the original image ROIs to be protected.

Transmorphing data insertion: Finally, the bitstream of the JPEG sub-image, along with some metadata, is inserted in the application segment marker 11 (APP11 marker) of the processed JPEG image. The sub-image can be protected by encryption with a key, e.g. the Advanced Encryption Standard (AES) or JPEG scrambling.[9]
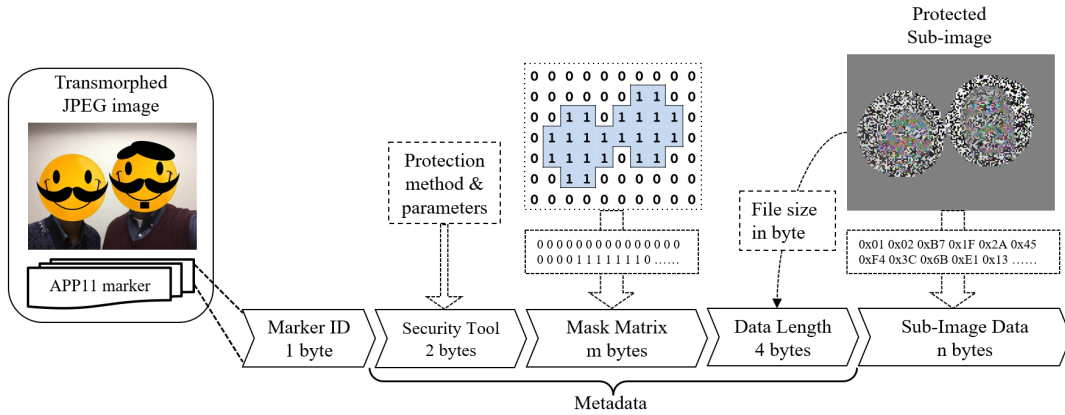

Figure 5. APP11 data in the transmorphed JPEG image file.[2]

The inserted metadata contains the following information: (i) the method used to secure sub-image, (ii) elements of the mask matrix and (iii) size of the sub-image in bytes. The elements of the Mask Matrix are encoded to a bitstream which is then written to another few bytes in the APP11 marker. The overhead added to the transmorphed JPEG image is mainly impacted by the size of the JPEG encoded sub-image, while the metadata overhead is negligible.

Transmorphing reconstruction (Re-transmorphing): The reconstruction procedure aims at recovering the original image from a transmorphed image, by reversing the above transmorphing protection operations: extracting protected sub-image and Mask Matrix from inserted transmorphing data, decrypting protected sub-image and replacing the obfuscated ROIs in transfomed image with corresponding information in the sub-image. Since the inserted Mask Matrix and sub-image preserve the complete information about the original image corresponding to the protected ROIs, the protected image is robust to most image transformations. In case the sub-image is protected, a key needs to be provided to decrypt or descramble the extracted sub-image. The replacement process can be done in either frequency (DCT coefficient) or spatial (pixel) domain depending on the transformation applied to the transmorphed image.

3.2 ProShare Framework

ProShare is a framework and an implementation designed by Yuan et al.[1] It is built based on a public key infrastructure (PKI) integrated with a cipher-text policy attribute-based encryption (CP-ABE).[1] In ProShare, a photo is securely protected by the secure JPEG transmorphing algorithm with a key.

ProShare is also the name of an application developed at Multimedia Signal Processing Group of EPFL that implements the above mentioned architecture and consists of two parts: (i) a mobile application for iOS and
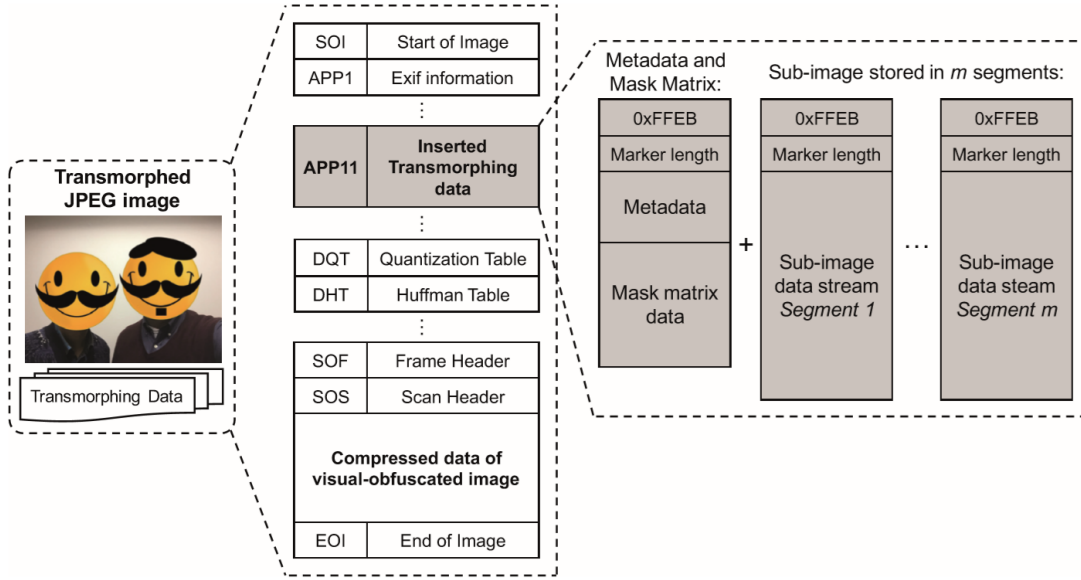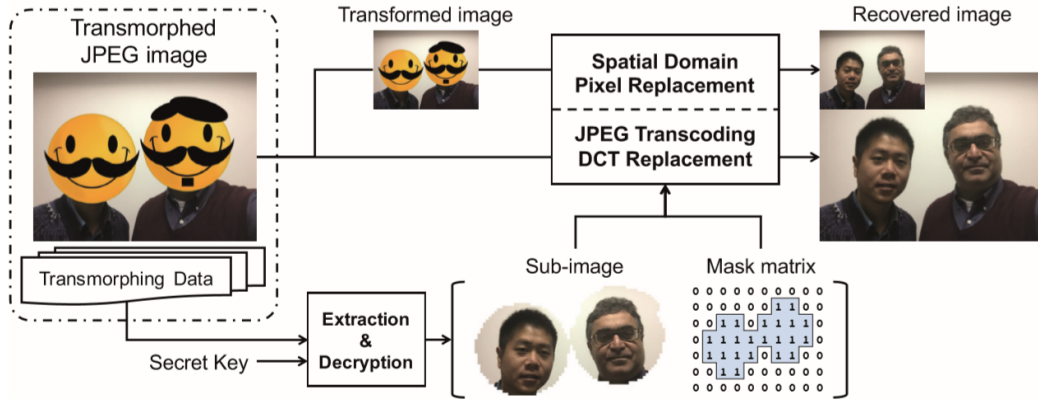
Figure 6. The syntax of Transmorphed JPEG image file.[2]

Figure 7. Transmorphing reconstruction.[2]

Android (available at AppStore* and Google Play Store†) and (ii) a web server hosting images and managing keys. In this paper, Android mobile application and web server will be used in order to design and implement secure image sharing for HDR images.

3.2.1 ProShare Android

ProShare for Android implements a user friendly set of features. *User login and registration:* in order to use ProShare App a user has to be registered. Each user registers an account using an e-mail address. Upon registration, two pairs of keys (TPK/TSK, APK/AMSK) are generated. *Friending and following:* An important feature of the ProShare App is the concept of trusting other users and to build or revoke friendships. In ProShare App User Profile Tab, users can search other users and follow or trust them. Followed users pictures will appear in Favorites Tab, but following does not mean that ProShare will show unprotected pictures of followed users. When a user A trusts another user B, a trusted group has to be chosen, this will allow the trusted user B to view the protected pictures of the user A when this user A selects to share a picture with the group in which
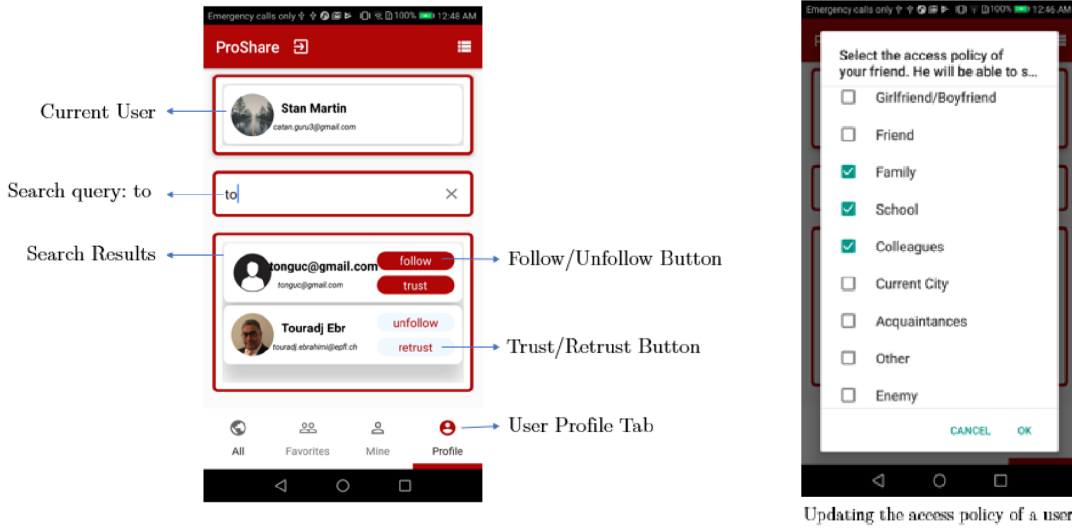

Figure 8. ProShare Android User Profile Tab screenshots

.

the user B is included. This relationships are stored in the server using CP-ABE. *Photo protection and sharing:* in ProShare Android, users can import images from device gallery or it can be captured from device camera. These images can be regular 8-bit JPEG or 360 panoramic images. ProShare App defines two different types of images: a public image that can be viewed by everyone and a private image that can only be viewed by groups of users chosen by the owner of that picture. *Photo accessing and viewing:* When a user A tries to view an image of another user B, ProShare App checks in the server if user A is included in the group that user B had trusted when sharing the photo. If it is the case, the private image will be displayed to user A, if it is not the case, the public image will be displayed to user A. In addition, the application also allows users to share images protected in Secure JPEG on Facebook in the form of App Links. By clicking on the shared link, other Facebook users are directed to either the ProShare App (if ProShare is installed on a mobile device) or a web interface. If the user has an account on ProShare, the photo can be shown to the user depending on his/her "relation" attributes with the owner of the photo.

ProShare Android allows users to edit both private an public images by different editing methods. This editing methods are: (i) adding stickers, (ii) adding bubbles, (iii) processing filters, (iv) add frames, (v) pixelation and (vi) inpainting.[10] Once editing process is finished, users choose the group of users allowed to view the private image, Mask Matrix is generated by comparing private and public images and computed mask and both images are uploaded to the server where JPEG transmorphing is performed.

3.2.2 ProShare Server

Android and iOS ProShare apps use the same ProShare server. This server consists of three components:[2]

- **Trusted Component (TC):** hosts some of the users' secret information such as Public Key Cryptography (PKC) private keys and ABE master keys. The Secure JPEG protection and recovery also happen on this component. This part of the web service is used to simulate some of encryption/decryption operations that are supposed to happen on the client device, which is assumed to be trusted but not yet implemented.

- **Key Service Component (KSC):** acts as the key server, for storing encrypted keys and issuing PKC public keys.

- **Content Service Component (CSC):** hosts the public images posted by all registered users.
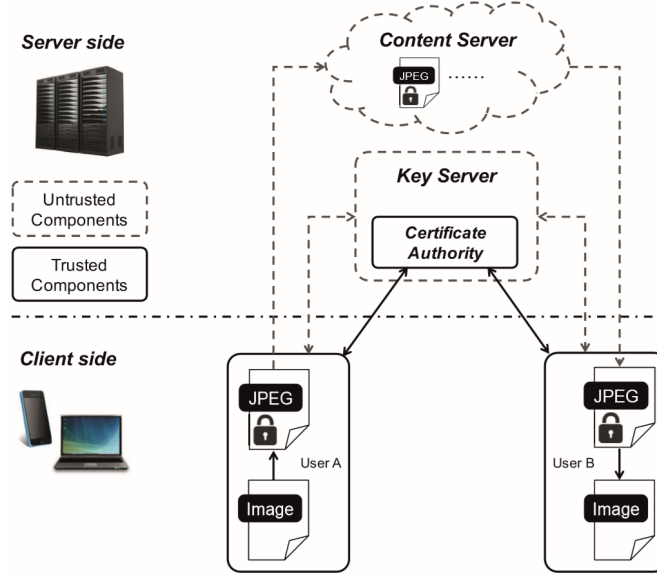
Figure 9. ProShare server architecture[2]

4. EXTENDING TRANSMORPHING TO HDR

This section explains how JPEG transmorphing algorithm has been modified in order to operate with HDR images encoded in JPEG XT and how HDR imaging using JPEG XT standard has been integrated into the framework of ProShare application.

4.1 Description of the Method

In this subsection, all the designed modifications to the JPEG transmorphing algorithm, ProShare Android and ProShare server are described.

The relationship between the different functionalities that were implemented in this work is shown in Figure 10. Transmorphing algorithm is located on the server side of ProShare framework; HDR acquisition, editing, sharing and displaying are functionalities that are implemented in the client application of ProShare framework. The server logic has been adapted in order to cope with HDR images. Further in this paper, a more detailed description of these functionalities is provided.

4.1.1 Transmorphing algorithm for HDR images

The JPEG transmorphing algorithm (explained in section 3) has been modified in this work. The modified algorithm is now called transmorphing algorithm for HDR images and is able to transmorph and recover HDR images encoded in JPEG XT format. The procedure of the algorithm is the same as the original JPEG transmorphing algorithm. The APP11 marker containing the HDR information of the original image is encrypted with the same key as the protected data and inserted into the transmorphed image. This process is illustrated in Figure 11. The HDR information can only be decoded if the image is recovered with the transmorphing algorithm for HDR.

The original JPEG transmorphing algorithm uses the APP11 marker for storing the transmorphing information. However, JPEG XT also uses the APP11 marker to store HDR information. This causes incompatibility issues since JPEG transmorphing algorithm is not able to recover an image if it contains more information than the transmorphing data in the same APP marker. Moreover, some JPEG XT HDR decoders are not able decode an HDR image if the APP11 marker contains additional data. In order to resolve this issue, the transmorphing data was stored in the APP8 marker instead of the APP11.

The transmorphing algorithm for HDR images is compatible with both JPEG XT HDR images and regular JPEG SDR images. The original image can be HDR or SDR, and the protected image must be SDR. The
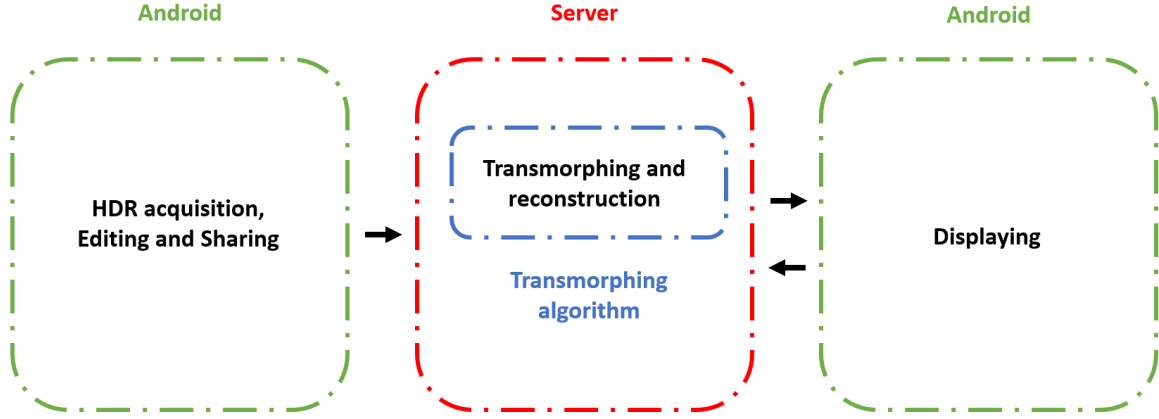
Figure 10. Diagram indicating the relationship between the parts of the ProShare framework that have been adapted in order to cope with HDR content.

transmorphed image is SDR and the recovered image (retransmorphed) is HDR or SDR, depending on the original image. In Figure 11, the transmorphing algorithm for HDR images is illustrated.
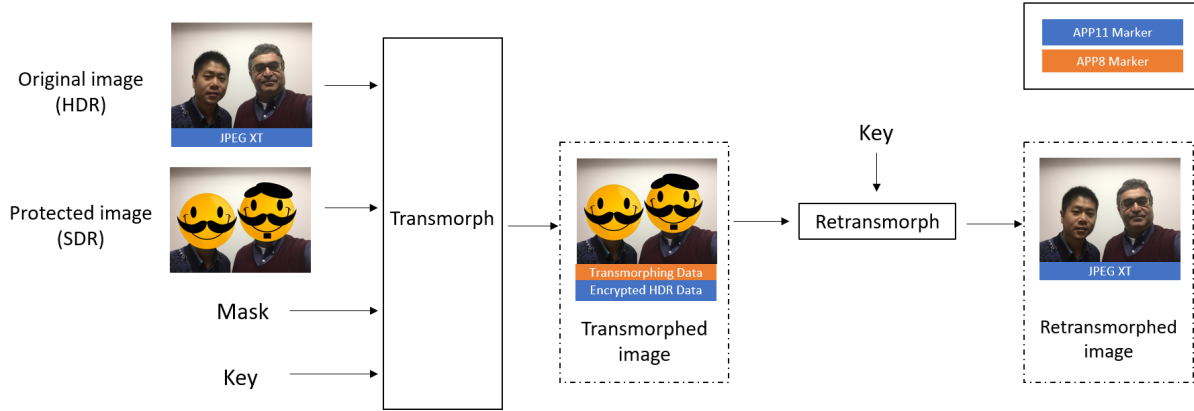

Figure 11. Transmorphing algorithm for HDR images. Retransmorph denotes transmorphing reconstruction.

Clarification of terminology: in the transmorphing algorithm there are four different images (original, protected, transmorphed and re-transmorphed), as illustrated in Figure 11. In the ProShare application, the images are only differentiated by the displayed information: original and re-transmorphed (recovered) images are called private images, and protected and transmorphed images are called public images.

4.1.2 ProShare Android

The ProShare Android application has been modified in this paper in order to include two new features: (i) sharing HDR images and (ii) displaying HDR images.

A new option in ProShare Android application has been designed in order to allow users to acquire and share HDR images. In the proposed new design indicated in Figure 12, the application interface includes a new option to share HDR images. When this option is selected, it displays a camera view, and when the shot button is clicked, the application captures 3 images with different exposures and merges them in order to create an HDR image. The HDR image is then encoded with JPEG XT. The encoded image is the private image, and the base JPEG layer is displayed in the application editing form. The resulting edited image is the public image. The mask is computed as a difference between the base layers of the private and the public images. The user has to choose the group of people that are allowed to view the private image, for which a key is creates. Finally, the images, the mask and the key are uploaded to the server.
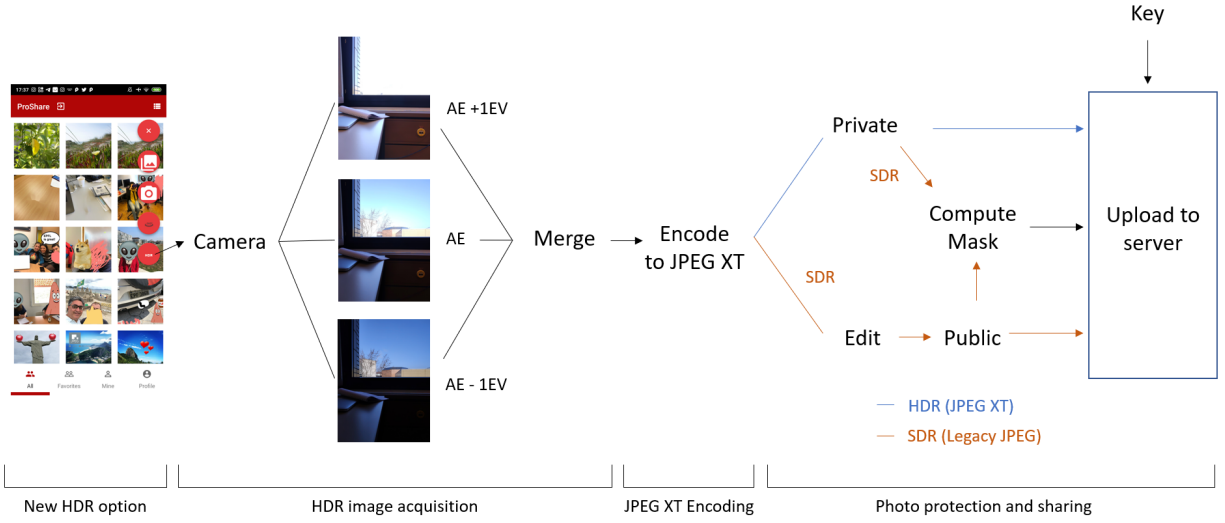
Figure 12. Designed process for sharing HDR images in ProShare Android.

HDR photo accessing and viewing: In the proposed design illustrated in Figure 13, when a user attempts to visualize an HDR image, the application checks if that particular user has access rights. If the user is not allowed, the protected image is displayed in SDR. If the user is allowed to view the private image, the application checks if the device where the application is running has an HDR display. If the device does not have HDR display capabilities, the recovered private image is displayed in SDR (only the base layer is displayed). If the the device has HDR display capabilities, the private image is decoded using the JPEG XT decoder and the image is displayed in HDR.
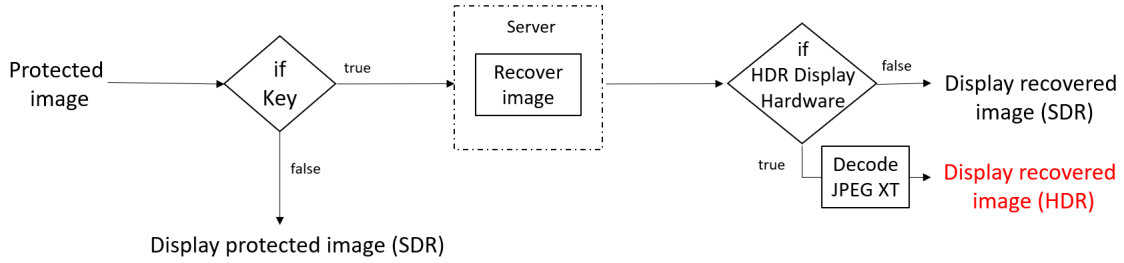

Figure 13. Design of the process for displaying HDR images in ProShare Android.

4.1.3 ProShare Server

The ProShare server has been modified to include the new transmorphing algorithm in order to be able to operate with HDR images. The architecture of the modified ProShare server is illustrated in Figure 14. The processing that takes place on the server side differs depending on whether the uploaded image is HDR or SDR. If an SDR image is uploaded, the server applies the previous version of JPEG transmorphing algorithm, whilst if an HDR image is uploaded, the server uses the new transmorphing algorithm for HDR images. The server database structure has been also updated with a new property that indicates if an image uploaded to the server is HDR. Based on this property, the server can make a decision on which transmorphing algorithm to engage for

re-transmorphing. Note that the same PKI system of the existing ProShare server is used in the updated server architecture.
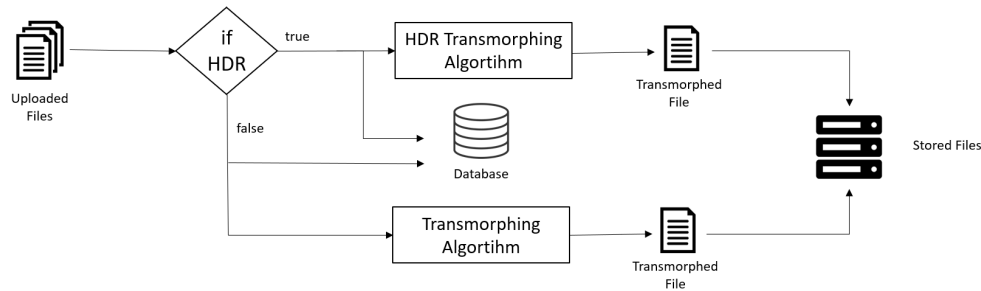

Figure 14. ProShare server logic to differentiate between HDR and SDR images.

4.2 Implementation

4.2.1 Transmorphing Algorithm for HDR images

The very first modification to the transmorphing algorithm that is required in order to incorporate JPEG XT is to change the application marker used to store transmorphing information in the JPEG file . The encrypted data were previously stored in the APP11 marker. However, JPEG XT uses the APP11 marker for storing HDR information. Thus, this raises a problem of incompatibility. In the previous existing system all images transmorphed by the algorithm in ProShare were encoded by Android or iOS legacy JPEG encoders. Now, however, with the new design, the public image is encoded using the Android JPEG encoder whereas the private image is encoded using the JPEG XT encoder. The proposed modified transmorphing algorithm takes as the input the original (SDR), the key and the mask, and produces the transmorphing data. Afterwards, the algorithm creates a copy of the protected (public) image that is called a transmorphed image. The transmorphing information is stored in the APP8 marker of the transmorphed image. As the final step, the original HDR APP11 marker is copied to the transmorphed image. However, if the APP11 of the private image is copied to the transmorphed image, the protected data is not fully secured since the part of the original image can be recovered without the key when decoding the JPEG XT. In order to solve this problem the APP11 marker is encrypted with the same key as the transmorphing information. When the image is recovered, the application decrypts the APP11 marker using the same key as for the transmorphing data.

4.2.2 Client

The previous version of ProShare Android client application has been modified in order to implement the option of sharing HDR images encoded in JPEG XT and the functionality to display HDR images. The developed features and improvements in the Android client can be divided in four groups: (i) HDR image acquisition, (ii) JPEG XT encoding, (iii) editing and sharing HDR, and (iv) displaying HDR.

HDR images are typically represented with more than 8-bit channels. Thus, the previously existing editing methods implemented in ProShare Android application cannot be directly applied. Additionally, the logic implemented in the editing process in ProShare application was previously based on encoding and decoding of JPEG files while applying modifications in the uncompressed domain. This logic cannot be used for JPEG XT HDR images due to the fact that JPEG XT encoding and decoding process is more time consuming when compared to the legacy JPEG, as the former is not optimized for the current hardware architecture. For these reasons, in the new option introduced in ProShare Android for sharing HDR images only the private image is HDR and it cannot be edited. The public image is SDR and can be edited using the previously developed editing features.

Figure 16 describes the logic of processing of a JPEG XT input file. The encoded JPEG XT file (1) is cnsidered by ProShare Android as the private HDR file. This image is decoded with the Android legacy JPEG
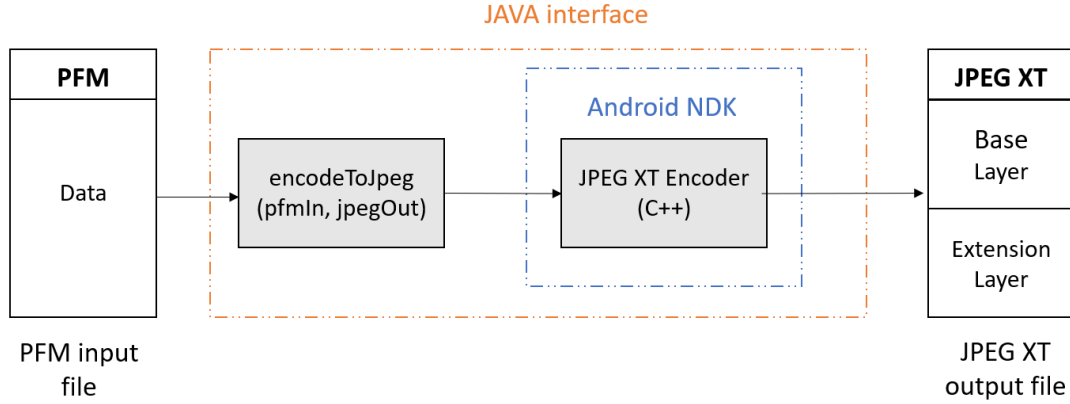
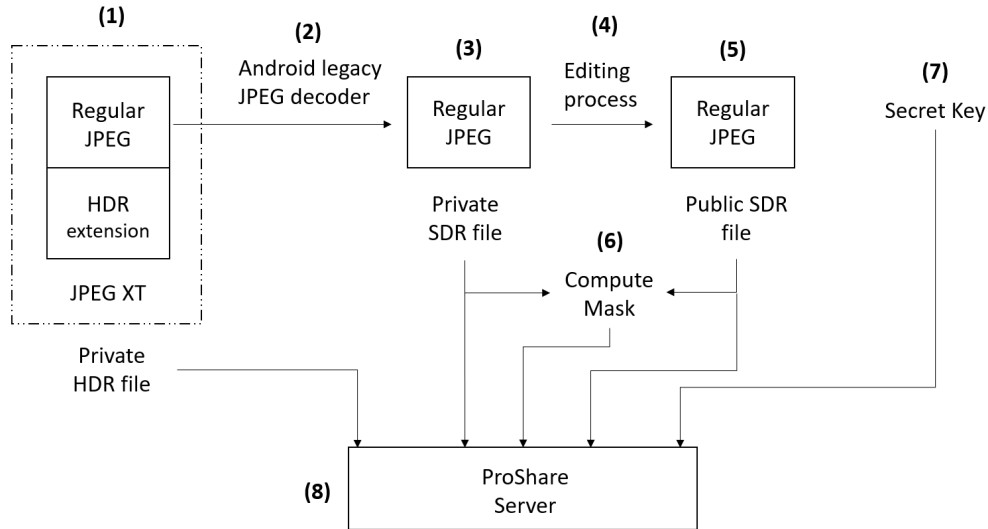Figure 15. JPEG XT encoding process implemented in ProShare Android.

Figure 16. ProShare client editing activity logic for an HDR picture.

decoder (2) (thanks to the fact that JPEG XT is backwards compatible with JPEG) and displayed on the Android device. The displayed image is encoded with the Android legacy JPEG encoder. As the result of the JPEG decoding, an SDR copy of the private file (3) is obtained. This private SDR file is further used by the transmorphing algorithm. Users can edit this private image (3) with all available editing methods in ProShare (4) in order to protect sensitive visual information such as faces. The edited image becomes a public image (5) which is also encoded with the legacy JPEG encoder before being transferred to the server. A mask file is created by comparing public and private SDR images. In order to share the images, a user has to choose the group of users allowed to view the private image and a secret key (7) will be generated. The private HDR and SDR files, the public SDR file, the mask file and the key are finally uploaded to ProShare Server (8).

During the work related to the findings presented in this paper, a simple standalone application called HDR Check was developed. This application that uses Android API `Display.HdrCapabilities` class to display the supported HDR standards by the device display (Figure 17). HDR Check has been installed on several Android

devices available in the Multimedia Signal Processing Group laboratory to check if any of them has an HDR display and what formats they support. Smartphones from 2016 and older do not support HDR. High-end smartphones from 2017, such as Samsung Galaxy Note 8, support HDR10, and cheaper phones from 2019, such as Xiaomi Mi 9 and Xiaomi Mi 9 Lite, support HDR10 and HLG. The most popular HDR format is HDR10, and almost every device that supports HDR supports HDR10. Thus, this is the format that was decided to be used in ProShare, to check if the device has an HDR display.
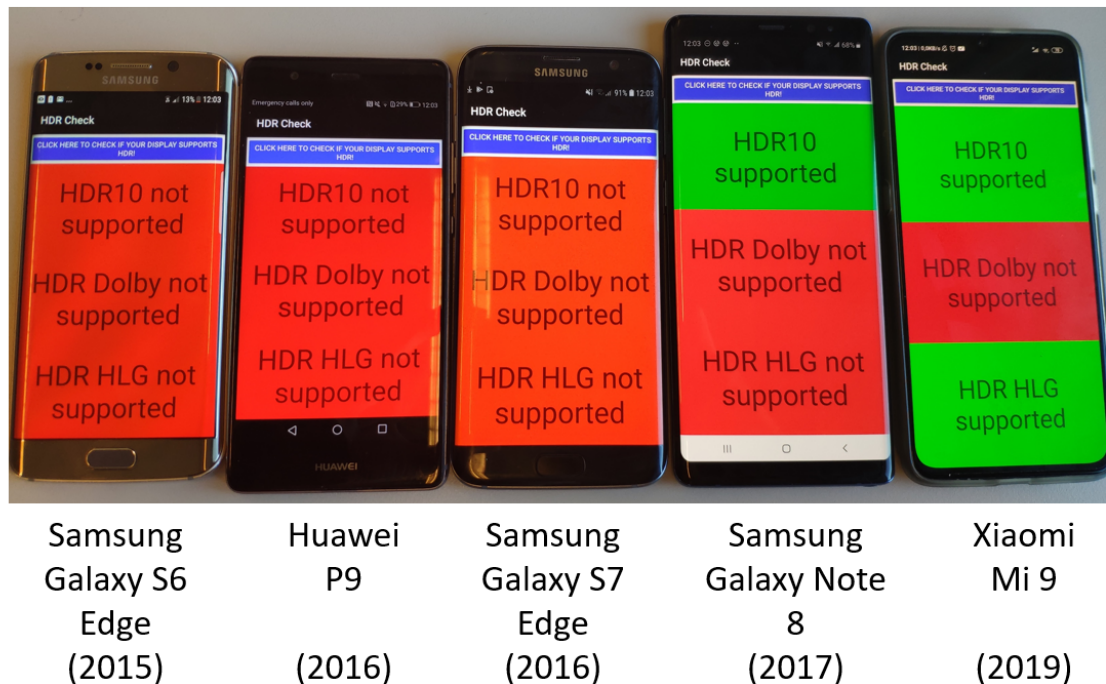

Figure 17. HDR display support checking with HDR Check application.

In this work, it was decided to use this tone-mapping functionality as a temporary solution until new Android architecture versions support more than 8-bit images. It is not required to have an HDR display in order to show a tone-mapped version of the original HDR image.

4.2.3 Server

The ProShare web server operates on an Apache HTTP Server. The programming language used is PHP. The server consists of three components: namely, Trusted Component ,Key Service Component, and Content Service Component. The new proposed transmorphing algorithm for HDR images has been introduced in the Trusted Component of the ProShare server in order to transmorph HDR images. At the same time, the original JPEG Transmorphing algorithm is also conserved in the Trusted Component.

A new property that indicates whether an image is HDR has been introduced in the ProShare server database. The original JPEG transmorphing algorithm is used for SDR images and the new transmorphing algorithm is used for HDR images. Once the image is recovered, it is sent to the client. With this modifications the ProShare server is able to protect, store and recover HDR pictures in JPEG XT format without a need for further modifications of the rest of the functional components of the server.

5. CONCLUSIONS AND FUTURE WORK

In this paper, the support for HDR imaging using JPEG XT has been implemented in the framework of ProShare application. This allows sharing privacy protected images in HDR. The JPEG transmorphing algorithm implemented in ProShare framework has been adapted to cope with HDR images represented in JPEG XT format. A

new functionality has been added to the ProShare client application to be able to capture, encode, edit and share JPEG XT HDR images. JPEG XT images can now also be decoded in the application. The ProShare server now integrates HDR support conforming to JPEG XT standard. This allows the ProShare client application to upload and receive protected HDR images.

The present work has left some issues that could be addressed in the future. HDR pictures cannot be displayed in HDR due to limitations of the Android architecture. However, the community has shown an increasing interest in HDR content, and it is not unrealistic to expect a full compatibility with HDR content directly implemented in Android API in the upcoming years. We have prepared the architecture in ProShare Android application to display HDR images. Only the tone-mapping view would need to be replaced with an HDR displaying view as soon as the HDR rendering will be fully supported by the Android API. Other further developments could address the optimization of the JPEG XT library in the Android framework. Currently, the reference JPEG XT library is used and it is implemented with Android NDK, which is not optimal. If JPEG XT is integrated in the Android API, it could improve the processing speed for the HDR images.

ACKNOWLEDGMENTS

REFERENCES

[1] Yuan, L. and Ebrahimi, T., "Image transmorphing with JPEG," in [*2015 IEEE International Conference on Image Processing (ICIP)*], 3956–3960 (Sept. 2015).

[2] Yuan, L. and Ebrahimi, T., "Image privacy protection with secure JPEG transmorphing," *IET Signal Processing* **11**, 1031–1038 (Aug. 2017). Publisher: IET Digital Library.

[3] Kunkel, T., Daly, S., Miller, S., and Froehlich, J., "Perceptual design for high dynamic range systems," in [*High Dynamic Range Video*], 391–430, Elsevier (2016).

[4] "ITU-R Rec. BT.2100-2: Image parameter values for high dynamic range television for use in production and international programme exchange," (2018).

[5] Diaz, R., Blinstein, S., and Qu, S., "Integrating HEVC Video Compression with a High Dynamic Range Video Pipeline," *SMPTE Motion Imaging Journal* **125**, 14–21 (Jan. 2016). Conference Name: SMPTE Motion Imaging Journal.

[6] "ST 2084:2014 - SMPTE Standard - High Dynamic Range Electro-Optical Transfer Function of Mastering Reference Displays," *ST 2084:2014* , 1–14 (Aug. 2014). Conference Name: ST 2084:2014.

[7] Boyce, J. M., Ye, Y., Chen, J., and Ramasubramonian, A. K., "Overview of SHVC: Scalable Extensions of the High Efficiency Video Coding Standard," *IEEE Transactions on Circuits and Systems for Video Technology* **26**, 20–34 (Jan. 2016). Conference Name: IEEE Transactions on Circuits and Systems for Video Technology.

[8] Richter, T., Artusi, A., and Ebrahimi, T., "JPEG XT: A new family of JPEG backward-compatible standards," *Ieee Multimedia* **23**(3), 80–88 (2016).

[9] Yuan, L., Korshunov, P., and Ebrahimi, T., "Secure jpeg scrambling enabling privacy in photo sharing," in [*2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*], **4**, 1–6, IEEE (2015).

[10] Upenik, E., Akyazi, P., Tuzmen, M., and Ebrahimi, T., "Inpainting in Omnidirectional Images for Privacy Protection," in [*ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*], 2487–2491 (May 2019).