

A Smartphone Application for the Creation of Legal Document Photographs

Master thesis

W.Diphooorn

Abstract—Legal document photographs have to conform to the requirements stated in the ICAO photograph guidelines. Due to the average individual being unfamiliar with these requirements, legal document photographs are normally taken by a professional photographer. This research is focused on offering an alternative to the photographer with a smartphone application. The application detects the conformance to all requirements using modern image processing and computer vision algorithms. The user is informed to which requirements the image does not conform and is instructed by on-screen gestures, text and vocal messages.

Index Terms—Image quality estimation, Semantic segmentation, International Civil Aviation Organization (ICAO), Biometrics, Legal Document Photographs.

I. INTRODUCTION

Taking a high quality photograph for legal documents like a passport, ID-card or driver license takes quite some time and effort. The requirements of the image quality attributes for such a photograph are easily met by the cameras of today's smartphones. However, besides the quality attributes (e.g. resolution, sharpness and dynamic range) for these photographs, there are other attributes to which the photograph has to conform. These attributes are described in the International Civil Aviation Organization (ICAO) photograph guidelines. There are several applications available for the creation of legal document photographs. However, most of these applications are focused on supporting professional photographers: the experts. These applications validate a captured photograph but provide no guidance in the process of capturing the photograph. The applications assume that the captured photograph already conforms to most requirements and only offer some basic image processing tools for subsidiary editing. While this might be convenient as a final verification for an expert, it is not very helpful for the average individual. A few applications are not intended to be used by experts, but the capability of these applications is limited. Most of these applications are simple cropping and fitting tools that helps the user to make the photograph conform to the geometrical requirements such as size and eye-distance. Most pose-specific and photographic attributes such as skin tone, hair across eyes, reflection of eyeglass lenses or shadows are not taken into account. The result: many rejected photographs.

For these reasons, most individuals still depend on a professional photographer for the creation of their legal document photographs. A more advanced application that instructs users in the process of creating a legal document photograph similarly to a photographer, could make a visit to the photographer superfluous. This makes the concept of such an application

very interesting. Therefore, the primary research question of this study is: how to develop an application that replaces the role of the photographer in the creation of a legal document photograph? This question introduces the following sub-questions. (1) What are the requirements of a legal document photograph, (2) How to test a photograph on the conformance to these requirements, (3) How to instruct the user on the non-conforming photograph characteristics?

A previous research already addresses the issue of testing photographs to the conformance to the ICAO standards [29]. The present paper aims to contribute to this previous research by improving the accuracy of these tests by using modern image processing techniques (e.g. deep learning). Moreover, because of the intention to provide an alternative to a professional photographer, the current paper implements these ICAO tests in a smartphone application. This application guides users in creating a legal document photograph with their smartphone camera.

The paper is organized as follows: Section II discusses related work of this research and provides insight on the requirements of a legal document photograph. Section III-A presents the approach on testing a photograph against requirements. Section III-B describes how feedback should be provided on non-conforming characteristics. Section IV is about the generic user interface and the global architecture of the application. Section V presents the results of this paper. Section VI discusses the results presented in section V. Finally section VII provides short summary of the paper.

II. RELATED WORK

A. publications on automated ICAO photograph tests

The paper by [8] introduced a benchmark tool for the performance evaluation of automated ICAO [29] photograph test software packages. For this benchmark, the ICAO guidelines were translated into 28 testable requirements. In the current paper some adjustments are made to these requirements. The first adjustment was dropping the requirement of an image not being ink marked or creased. This requirement is irrelevant for this study since this paper is focused on the creation of a legal-document photograph rather than testing existing photographs. The second adjustment was merging the non-varied background and no object in background requirement since the discrepancy between these requirements is hard to determine and both requirements require the user to change the background. The remaining 26 requirements can be found in Table 1 along with the requirements: 26, 27 and 29 which are introduced in the current paper.

The paper by [8] devised an image validation method for every requirement. The benchmark score for every method was compared to the score of two commercial software packages. The accuracy of proposed method varies between 99.4% and 78.4% depending on ICAO characteristic that is tested. Overall the methods proposed by [8] performed superiorly in comparison to the two commercial software packages.

Next to that, the company Vsoft published two papers that continue the work of [8] in collaboration with the Federal University of Paraiba and the Federal University of Campinda Grande. The first paper [2] presents an innovative approach for three of the image validation methods that were introduced by [8]: unnatural skin tone, flash reflection on skin and shadow across face. The second paper [21] presents an innovative method for the image validation of pixelation, hair across eyes, veil over the face and opened mouth.

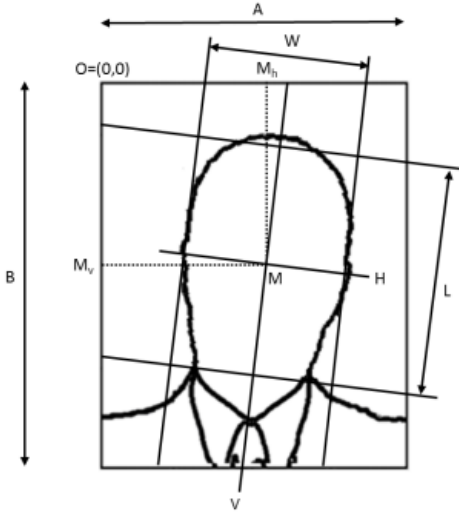


Fig. 1: Camera coordinates (image source [29])

B. Related software applications

As mentioned before, there already exist a few applications that can verify the conformance of a photograph to the ICAO guidelines. It is important to study these applications to ensure the additional value of this research.

1) *The BioLab-ICAO Benchmark*: The BioLab ICAO benchmark is developed by the university of Bologna and made available on the FVC-ongoing website [7]. The benchmark is developed for the validation of ICAO test software. The dataset used for the benchmark contains 5588 images of which 720 are available for local testing and training of algorithms.

2) *photomatic*: Photomatic is a web-based application. The program gives a score between 0-100% for the requirements listed in the work of [8], an image needs a score of above 50% in order to comply to a certain requirement. The application takes a single image as its input. Therefore, feedback can only be provided after capturing the photograph.

TABLE I: Requirements of a legal document photograph

Geometrical characteristics

- 1 Eye distance (min 90 pixels)
- 2 Vertical position $[0.3B \leq M_v \leq 0.5B] * f_{ig1}$
- 3 Horizontal position $[0.45A \leq M_h \leq 0.55A] * f_{ig1}$
- 4 Head image width ratio $[0.5A \leq W \leq 0.75A] * f_{ig1}$
- 5 Head image height ratio $[0.6B \leq L \leq 0.9B] * f_{ig1}$

Photographic characteristics

- 6 Blurred
- 7 Unnatural skin tone
- 8 Too Dark/Light
- 9 Washed out
- 10 Pixelation
- 11 Flash reflection on skin
- 12 Red eyes
- 13 Shadows across face
- 14 Shadows behind head
- 15 Dark tinted lenses
- 16 Flash reflection on lenses

Pose-specific characteristics

- 17 Looking away
- 18 Hair across eyes
- 19 Eyes closed
- 20 Roll/Pitch/Yaw Greater than 8°
- 21 Frames too heavy
- 22 Frames covering eyes
- 23 Hat/Cap
- 24 Veil over face
- 25 Mouth open
- 26 Non-neutral expression

Scene-specific characteristics

- 27 Wrong background colour
- 28 Varied background / Object in background
- 29 Multiple faces in image.

3) *Passport Photo Editor*: Passport Photo Editor is an Android application that claims to guide the user in the process of creating a legal document photograph. The application helps the user with cropping and fitting the image so that it conforms to the geometrical requirements. The application also detects some pose-specific attributes (e.g. closed eyes and smiling) but fails on detecting attributes such as skin tone, hair across eyes, reflection of eyeglass lenses, shadows. Resulting in many of these photographs being rejected by authorities.

4) *ID Photos Pro 8*: ID Photos Pro 8 is a Windows application that guides a user in the process of creating a legal document photo from an existing photo. It detects a few of the requirements listed by [8] such as facial expressions, reflections and shadows. The application however cannot provide real-time feedback on those aspects since it uses a static photograph. It gives a warning of the associated issue and ask the user to provide a different photograph. The application only addresses one issue at a time. A user that is not familiar with the ICAO requirements might need many retakes to get a photo that conforms to the requirements.

III. METHODS

A. Image validation methods

This paper introduces an extended list of the requirements derived by [8]. As stated by the papers [8] and [10], the translation of the ICAO guidelines into testable requirement involves some human interpretation. Even experts do not always seem to agree on the definition of a conforming photograph. After studying the ICAO guidelines, it was decided that the list of requirements was incomplete and therefore, requirements: 26, 27, 29 (Table 1) are added to list of requirements.

Since the requirements of Table 1 are based on the work of [8], the paper of [8] already provides an algorithm that validates the conformance for most of the requirements. However, their approach is not intended for an application that provides real-time feedback. Therefore, some algorithms might be too slow for the present paper's real-time application. Since the paper by [8] is over eight years old, newer techniques are available, which could be beneficial for the speed of the application. Moreover, these techniques have the potential to improve the accuracy of the algorithms, which were already not that accurate, as stated in the conclusion of paper [6].

The methods for the validation of requirement: 7, 8, 9, 10, 11, 13, 14, 16, 21, 28 are based on the work of [8], [2] and [21] but have been slightly modified. The implementation of the image validation method of requirement: 6, 12, 15, 17, 18, 19, 20, 22, 23, 24, 25, 26, 27 and 29 is not directly related to the work of [8], [2] or [21]. The descent of these methods will be explained in the next section.

Segmentation masks

A segmentation mask is used to differentiate hair, skin and background pixels (see Figure 2). These masks are created using a deep neural network. The architecture of this network is an agglutination of the MobileNets[9] and the ENet[22] network. This network is implemented in Python and trained using the images from [11] with the open source neural network library Keras.



Fig. 2: Hair/Skin segmentation mask

The same approach is used to create a segmentation mask for glasses (see Figure 3). The segmentation model for glasses is trained using the celebA dataset [15] with the manually-annotated facial feature masks provided by [3].



Fig. 3: Glasses segmentation mask

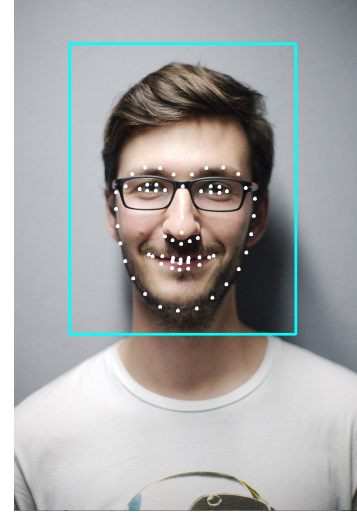


Fig. 4: Facial landmarks used by the algorithm

A.1. Geometrical characteristics

For many of the ICAO requirements it is important to determine the position of the eyes, nose and mouth. The paper [8] used a different technique for finding the position each of these facial attributes. In this paper the location of these facial attributes are estimated using the facial landmark detection algorithm of [12]. The algorithm returns the position of 68 facial features, see 4 for illustration of the facial landmark points. With the position of every facial landmark known, the verification of the geometrical characteristics is just a matter of measuring the distances between the points and comparing them with the requirements. This approach is also used to automatically crop the image to the size requirements of a legal document photograph (see Figure 4).

A.2. Photographic characteristics

6) *Blurred*: A sharp images is expected to have sharper edges than a blurred image. These sharp edges have a high spatial frequency which can be filtered out using a second derivative operator such as the Laplacian operator. The Laplacian operator is therefore very useful to the detection of blurred images. The sharpness of the image is measured using the method of [23]. This method can be described using the following equation:



Fig. 5: Natural skin pixels within the contours of the facial landmarks

$$Blurred = \frac{1}{N * M} \sum_m \sum_n [|L(m, n)| - \bar{L}]^2 < T \quad (1)$$

where

$$\bar{L} = \frac{1}{N * M} \sum_m \sum_n |L(m, n)|$$

and

$$L = I(m, n) \otimes \frac{1}{6} \begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

and where T is the threshold value for which the image is considered sharp.

7) *Unnatural skin tone*: The method proposed by both [8] and [2] are based on the work of [1]. The paper [1] describes the range of skin pixels within the YCbCr color-space. However, the work [2] obtained better results by adjusting the Y-channel range from $Y > 80$ to $70 < Y < 180$. Both the methods presented in [8] and [2] estimate the compliance to this requirement based on the percentage of neutral skin pixels within a rectangular region around the face. The skin segmentation mask (see Figure 2) provides a more accurate representation of the location of the skin pixels than a rectangular region around the face since this rectangular includes skin hair and facial accessories. The skin segmentation mask (see Figure 2) is not sensitive to an unnatural skin tone, the compliance to this requirement is estimated based on the percentage of neutral skin pixel within the skin segmentation mask. Following the adjustments of [2] on the work of [1], a pixel is considered to be a neutral skin pixels whenever it satisfies to 2, 3 and 4 where Y, Cb, Cr = [0, 255].

$$70 < Y < 180 \quad (2)$$

$$85 < Cb < 135 \quad (3)$$

$$135 < Cr < 180 \quad (4)$$

If enough pixels in the ROI are labeled as natural skin pixels, it is assumed that the image conforms to this requirement.

8) *Too Dark/Light*: The V-channel of the HSV color space describes the brightness of each pixel within an image and is therefore really useful for detecting images that are too dark/light. To limit the influence of the background color a rectangular region around the facial landmarks is used as the region of interest (ROI). The image is tested on this requirement using the average pixel intensity of the ROI and a lower and upper threshold value for which the image is considered to be too dark/light.

9) *Washed out*: The dynamic range of a washed out image is smaller in comparison to normal image. Therefore, a washed out image is detected by analysing its dynamic range. The image is considered to be washed out when it meets equation 5.

$$Washedout = \frac{\max(I) - \min(I)}{255} < T \quad (5)$$

where $\max(I)$ and $\min(I)$ are the minimum and maximum pixel intensity within the gray-scale image I, 255 the maximum intensity value of the image and T the threshold value for which the image is considered to be washed out.

10) *Pixelation*: An pixelated image contains several perfectly horizontal/vertical edges. The approach of this paper is similar to the work of [8]. Firstly, edges in the image are detected by applying a Canny edge detector operation on the image. Secondly, the percentage of horizontal/vertical edges is estimated with a Hough transformation. when the image contains a lot of perfectly horizontal/vertical edges, the image is considered to be pixelated.

11) *Flash reflection on skin*: The Y-channel of the YCbCr color space describes the luminance of every pixel within the image. Therefore, flashes on the skin are detected based on the number of high intensity pixels in the Y channel within the face region. The face region is defined using the facial landmarks as shown in Figure 5.

12) *Red eyes*: There are two challenges in finding a red-eye artifact. Firstly, determining the location of the eyes within the image. This location is found using the landmarks associated with the eye (see Figure 9). Secondly, defining the characteristics of a red-eye artifact. Different methods were considered for the detection of red pixels; however, most did not account for the luminance which resulted in flash reflections within the pupil being falsely labelled as a red-eye artifact. The work of [25] addressed this issue by using both a red and luminance pixel score for the detection of red eyes (see equation 6, 7 and 8). Using these equations and the landmarks associated with the eye, the presence of a red-eye artifact is estimated based upon the average red-eye score (see equation 8) within the region encircled by the eye landmarks.

$$Redness(x) = R(x) - \frac{(G(x) + B(x))}{2} \quad (6)$$

$$Luminance(x) = 0.25 * R(x) + 0.6 * G(x) + 0.15 * B(x) \quad (7)$$

$$Redeyes(x) = \max(0, 2 * Redness(x) - Luminance(x)) \quad (8)$$

where R,G,B are the channels of the RGB color-space.

13) *Shadows Across face*: The skin segmentation mask is not that susceptible for shadows, only shadow regions with an extremely low luminance will result in a gap within the mask. The probability of a shadow being present in the face region of the image is estimated based on the number of shadow pixels within the skin mask. Shadow pixels are defined as described in equation 9.

$$Shadow(x) = \begin{cases} 1, & \text{if } v_{skin}Y - \sigma_{skin}Y > x \\ 0, & \text{otherwise} \end{cases} \quad (9)$$



Fig. 6: Shadow behind head

14) *Shadows behind head*: The work of [8] shows that shadows can be detected by analysing the in X channel of the XYZ color-space. Shadows behind the head can therefore be detected by performing this analysis on the image background. Where the image background is defined as the region of the image that is not part of the skin, hair or glasses segmentation mask. The presence of a shadow is estimated contingent upon the number of pixels with a low intensity in the X-channel.

15) *Dark tinted lenses*: By making use of the skin segmentation mask (Figure 2) an accurate estimation can be made on the average skin colour of an individual. Whenever an individual is wearing transparent glasses, the colour of the skin pixels within the glasses segmentation mask (Figure III-A) will deviate from the colour of the skin pixels that are not part of this glasses segmentation mask. Dark tinted lenses are detected by comparing colour of the skin within those two mask as described with 10 and 11.

$$Tinted(x) = \begin{cases} 0, & \text{if } v_{skin} - (\sigma_{skin} * 2) > x \\ 0, & \text{if } v_{skin} + (\sigma_{skin} * 2) < x \\ 1, & \text{otherwise} \end{cases} \quad (10)$$

where v_{skin} is the average and σ_{skin} the standard deviation of the pixels within the skin mask.

$$Tinted = \frac{\left(\sum_m^M \sum_n^N f(m,n) * mask_{glasses}(m,n) \right)}{\sum_m^M \sum_n^N mask_{glasses}(m,n)} < T \quad (11)$$

16) *Flash reflection on lenses*: As mentioned before, for the detection of a flash on the skin, the luminance value of each pixel is found in the Y-channel of the YCbCR color space. A flash reflection on the lenses is detected based on the number of high-intensity pixels in the image region defined by the glasses segmentation mask (Figure 3).

A.3. Pose-specific characteristics



Fig. 7: Gaze estimation

17) *Looking away*: An individual is considered to be looking away whenever the location of its pupils deviates to much from the eye center. The position eye center can easily derived from the facial landmarks by averaging the location of all landmarks associated with the eye. However, determining the location of the pupil is a more challenging task. Several methods were considered such as gradient based pupil localisation (as proposed by [28]) and iris localisation based on the Hough transform (as proposed by [4]). However, the performance of these methods were quite poor for lower quality images (e.g. Pixelated, Washed out). Therefore an alternative method was devised that also works well with these lower quality images. This method is based on the blob-detection algorithm described in Algorithm 1. This image characteristic is evaluated by setting a maximum distance for which the position of pupil can deviate from the eye center.

Algorithm 1 Gaze estimation

```

contours ← findContours { [26] }
for each contour do
    if area > .30 * ROI and area ≤ .60 * ROI then
        perimeter ← arcLenght(contour)
        circularity ← 4 * π * area / (perimeter * perimeter)
        if circularity > 0.1 then
            pupils.append(contour)
        end if
    end if
end for
centroid ← mean(P1..P6) {Figure 9}
for each pupil do
    dist ←  $\frac{\text{euclidean}(\text{pupil}, \text{centroid})}{1/2 * \text{euclidean}(P1, P4)} * 100$ 
    distances.append(dist)
end for
return min(distances)

```

Where the contours are retrieved from the binary image as proposed by [26]

18) *Hair across eyes*: Because of the skin segmentation mask (Figure 2) and the facial landmarks (Figure 4) the position of both the hair and the eyes are known. This image characteristic is therefore evaluated simply by checking whether there are any hair pixel present in the eye region. The



Fig. 8: Hair across eyes

eye region is defined as a rectangular region around the eye landmarks with a padding of $\frac{1}{20}$ the interpupillary distance as described in the ICAO standards [29].

19) *Eyes closed*: This characteristic is evaluated based on the height/width ratio of eyelid landmarks (see Figure 9 and equation 12). Although there several different method for the detection of closed eyes, this method is most convenient due to the availability of the eyelid landmarks.

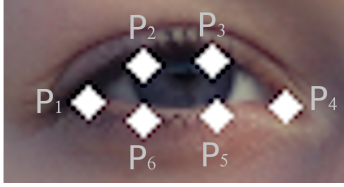


Fig. 9: Landmark points associated with the eye

$$EAR = \frac{||P_2 - P_6|| + ||P_3 - P_5||}{2||P_1 - P_4||} < T \quad (12)$$

20) *Roll/Pitch/Yaw Greater than 8°* : The head-pose estimation is based on the work of [17]. The pose of the face is estimated by calculating the translation and rotation between the facial landmarks obtained from the [12] algorithm and the same point in a frontal view of a generic three-dimensional face model.

21) *Frames too heavy*: The glasses segmentation mask (see Figure 3) only provides information on the outer contours of glass frame. Because of this, the following method has been devised for the detection of a heavy glass frame. First

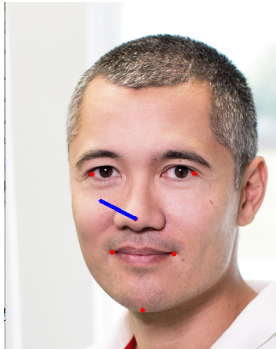


Fig. 10: Head orientation



Fig. 11: Eyeglass frame segmentation

an eroded version of glasses segmentation mask is subtracted from the original mask. Secondly, the number of skin pixels within the resulting mask are enumerated (based upon equation 10). Lastly, the thickness of the frame is estimated based upon the ratio of skin / non-skin pixels (see figure 11).

22) *Frames covering eyes*: Because of the facial landmarks the location of the eyes is known (see Figure 4). An occlusion of the eyes can therefore be easily detected based on the number of eyeglass frame pixels in the eye region (see Figure 9 and Figure 11). The eye region is defined as a rectangular region around the eye landmarks with a padding of $\frac{1}{20}$ the interpupillary distance as described in the ICAO standards [29].

23) *Hat/Cap*: Previous work tried to evaluate this characteristic using traditional computer vision techniques. However, this is a challenging task due to the large variation of hat/caps and hair styles. Therefore, Instead this characteristic is evaluated using a Convolutional Neural Network (CNN). The hat/cap classifier of this paper is trained with the MobileNetsV2 [24] network on a data-set of 30.000 images. These images show the forehead of individuals with and without headgear and originate from the [15] data-set. Several networks were considered but the MobileNetsV2 network was selected due to its great performance on mobile devices.



Fig. 12: Detection of a non-skin attribute within the face region

24) *Veil over face*: The skin/hair segmentation masks are quite accurate, a non hair/skin attribute like a veil can therefore easily be detected by analysing these masks within the face region. The face region is defined as the outer contours of the facial landmarks (see Figure 12).

25) *Mouth open*: As shown in Figure 4 the algorithm from [12] provides several landmarks associated with the mouth. Using these landmarks, it is easy to make a distinction between different mouth movements. An open mouth is detected using a mouth-aspect-ratio (MAR) [14]. The MAR is calculated following equation 13 using the points shown in Figure 13.

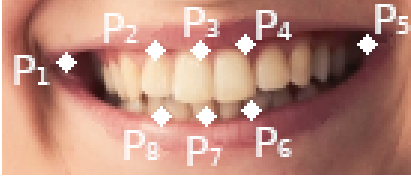


Fig. 13: Facial landmarks associated with the mouth

$$MAR = \frac{||P2 - P8|| + ||P3 - P7|| + ||P4 - P6||}{3||P1 - P5||} \quad (13)$$



Fig. 14: Vectors used for the classification of different emotional states

26) *Non-neutral expression*: The ICAO guidelines clearly state that the individual in the legal document photograph should have a neutral facial expression. As suggested by [20] the facial expression of an individual can be estimated by the relative position of the eyebrow and mouth landmark points (see Figure 14). A non neutral expression is detected using a linear support vector machine (SVM) classifier. This classifier is trained with the facial expression image data set by [27] and [16]. Since only neutral face expressions are allowed, the classifier only makes a distinction between neutral and non-neutral facial expressions.

A.4. Scene-specific characteristics



Fig. 15: Background mask (original image source [18])

27) *Wrong background colour*: The paper by [8] did not take this requirement into account since it is not part of the international ICAO guidelines. Many countries however state that the background colour should be white, gray or light

blue. Assuming that the photograph has a plane background (satisfies requirement 29). The background colour can be scrutinized by calculating the mean pixel intensity value of the background pixels. The background detection used by previous work was based on the watershed algorithm. After some experiments with this method, it was proven to be unsuitable for a real-time application. Therefore, an ad hoc solution was devised for the detection of backgrounds. In the case that the image complies to requirement 24 (Veil covering face) and 11 (flash reflection on skin), pixels that are not within the glasses, hair or skin mask should either be part of the background, an object in the background or a person's shoulders. Due to the characteristics of a legal document photograph, shoulder pixels are only present in the lower part of the image. Considering this, an approximation of the background region is made using the following method. Firstly, the image height is cropped so that only the upper 80% remains. Secondly, the glasses, hair and skin mask are merged into a single image mask. Thirdly, a dilation operation is applied to fill gaps within the segmentation mask. Lastly, the background area is obtained by subtracting this dilated mask from the cropped image (see Figure 15).

28) *Varied background / Object in background*: As suggested by [8], varied backgrounds are detected as described in [19]. A Prewitt operator is used to detect edges in the image. The background is assumed to be plain when no edges are detected within the background region of the image. The background region is defined as described earlier in this paper for the image validation of a wrong background colour.

29) Multiple faces in image

A Histogram of Oriented Gradients (HoG) face detection algorithm is used for the allocation of the facial landmarks (see [12]). Therefore, this characteristic can be easily evaluated based on the number faces that were detected by the face detection algorithm. The photograph is considered to conform to the requirement if the number of detected faces is equal to one.

B. User feedback

The user of the application has to be informed on whether the image conforms to the standards. When a camera frame does not conform to one of the standards, the application provides feedback to the user. This feedback informs the user on why the image does not conform and how he/she should behave in order to make the image conform. The user of the application and the individual of which the image is taken can be the same person, but this is not mandatory. In this section, the individual in the image is referred to as the "target".

B.1. Geometrical characteristics

The geometrical characteristics depend on the position of the camera in relation to the target. A smart cutting and cropping algorithm can take a subsection of the image that conforms to the geometrical characteristics. This relaxes the need of the target being perfectly centred in respect to the

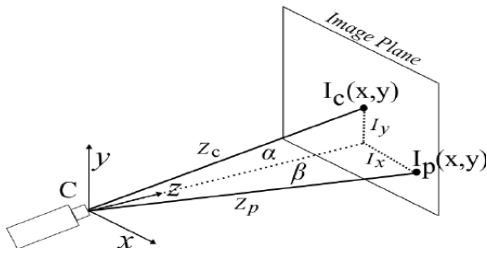


Fig. 16: Camera coordinates (image source Ferat Sahin)

viewing angle of the camera by increasing the Z-distance (Figure 16). However, there is a maximum distance, due to requirement I-1 and I-11. When the misplacement of the target with respect to the camera center in the X-Y directions is too large, the camera has to move towards the target in the X or Y direction. Indication lines show the desired movement of the camera.

B.1 Photographic characteristics

Prioritizing characteristics: It would be very inconvenient for the user if the feedback was provided on all the characteristics simultaneously. The application shows the ICAO test result for every requirement from section II but only provides feedback on one requirement at a time.

face detection: Many detection algorithms used in this approach depend on the detection of the face of the target. Whenever this face detection algorithm fails, the application should inform the user that no face has been detected. In such situations, the user should be provided with extra information, preferably with a list with the most common causes.

6) *Blurred:* The most common cause of a blurred image is a relative movement of the camera in respect to the target. Whenever this occurs the application asks the user to hold the camera still and make sure that the target is not moving.

7) *Unnatural skin tone:* An unnatural skin tone is a result of inappropriate lighting conditions. Therefore the application asks the user to adjust the lighting conditions accordingly.

8) *Too Dark/Light:* Whenever the image is too dark or light, there is probably a problem with the lighting conditions of the scene. The application asks the user to adjust the lighting accordingly.

9) *Washed out:* Washed out images are normally the artifact of a hazy lens, the application asks the user to clean the camera lens.

10) *Pixelation:* resizing the image might cause pixelation. If possible accordingly to the geometrical characteristics, the Z-distance between the camera and the target should be decreased. If this does not solve the problem, the hardware of the camera is incompetent. When multiple camera's are available, the application suggest swapping camera's. If no camera with a sufficient resolution is available, the application informs the user that the devices is unsuitable.

11) *Flash reflection on skin:* The application can turn off the flash of the camera. If the problem remains, there is an issue with the lighting conditions. This is resolved by informing the user on adjusting the brightness or redirecting of the scene's lighting.

12) *Red eyes:* The red-eye effect is caused by the reflection of the camera flash in the eyes of the target. In case of a red-eye artifact being detected, the application will ask the user to turn off the flash of the smartphone camera.

13) *Shadows Across face:* Shadows are caused by non uniform lighting. Whenever shadows are detected, the application informs the user about the lighting conditions for a legal document photograph.

14) *Shadows behind head:* see I-13.

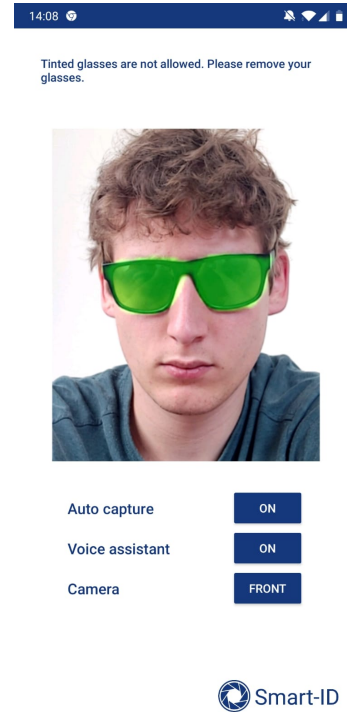


Fig. 17: Tinted glasses feedback

15) *Dark tinted lenses:* The application informs the user that dark tinted lenses are not allowed and asks to take off the glasses (see Figure 17).

16) *Flash reflection on lenses:* see I-11

B.3 Pose-specific characteristics

17) *Looking away:* The application asks the user to make sure that the target looks into the direction of the camera.

18) *Hair across eyes:* The application should indicate the occlusion and asks the user to remove it from the scene. The occlusion is indicated by a coloured rectangle.

19) *Eyes closed:* When the eyes are not entirely opened, the application will ask the user to open the eyes.

20) *Roll/Pitch/Yaw Greater than 8°:* The application indicates the desired orientation of the target using indication lines. See Figure 18 for an illustrative clarification.

21) *Frames too heavy:* The application indicates that the frame of the glasses is too heavy, application asks the user to take off the glasses.

22) *Frames covering eyes:* The application indicates that the frame is occluding the eye's and asks the user to adjust the position of the glasses accordingly.



Fig. 18: Face orientation feedback

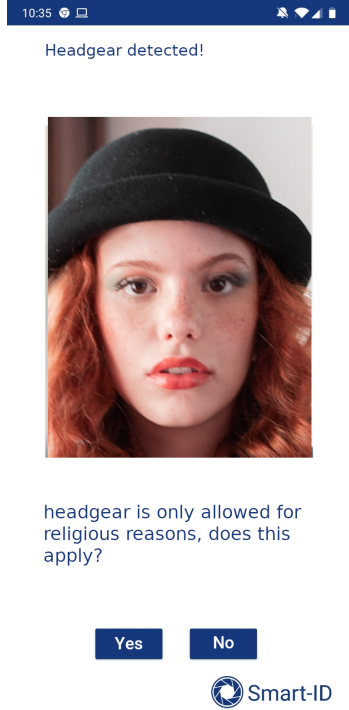


Fig. 19: Hat/Cap feedback

23) *Hat/Cap*: The application informs the user about the ICAO rules on headgear (see Figure 19). The user is given 2 options, first to take off the headgear. Second if worn for religious reasons, ignore the requirement. By giving the user the option to ignore this requirement, individuals that wear headgear, because of religious reasons are still able to use the application.

24) *Veil over face*: The application indicates the detected Veil with a coloured border and asks the user to remove the object.

25) *Mouth open*: The application asks the user to make sure that the mouth of the target is closed.

26) *Non-neutral expression*: The application informs the users on the regulation of non-neutral facial expressions in legal document photographs.

B.4 Scene-specific characteristics

27) *Wrong background colour*: The application informs the user about the permitted background colours and ask the user to change the background.

28) *Varied background / Object in background*: The application informs the user on the requirements of a plain neutral background and asks the user to change the background of the scene.

29) *Multiple faces in image*: The application informs the user that not more than one individual should be visible in the image.

IV. APPLICATION DEVELOPMENT

This section describes the global design of the application and the choices that led to this design.

As mentioned earlier, the paper [8] introduced a benchmark tool for the evaluation of image validation methods that test a photograph on the conformance to the ICAO [29] requirements. By using this benchmark tool, the development process actually consists of the development of two different applications. The benchmark tool namely only works with a Windows 32 bit executable. However, these kind of applications exclusively run on the Windows operating system.

Due to the availability of several Android smartphones for this research, the research focuses on the development of an application for Android operating system. Both the Android and Windows 32 bit application use the same image validation methods and it is therefore, desirable that the implementation of the image validation methods can be integrated in both applications.

An overview of the most common used programming languages for the development of an Android application is given in II. For each language a score is given for the average execution time, the availability of popular open source image processing libraries (e.g. Opencv, Dlib) and the support of the Android application programming interface (API). The score of the average execution time is based on the research of [5].

TABLE II: Programming language comparison

Language:	Time	IPCV libs	Android API	Supports Win-32
Java	+	+	++	No
C++	++	++	+	Yes
C#	+	+	+	Yes
Python	- -	++	-	No

Unfortunately, a program written in the Python or Java language can not be compiled into a standalone Windows 32-bit application. Therefore, the application should either be written in C# or C++ to be able to submit to the benchmark tool.

Because the image validation methods have to validate the images within an image stream, it is important that the execution time of these image validation methods is low to achieve an acceptable frame rate. This would make C++ the

most suitable language for the implementation of the image validation methods.

However, for the development of the Android application Java is the preferred language. Due to Java being the official language of the Android operating system and therefore it has the best support of the Android application programming interface (API).

Fortunately, it is possible to call C++ functions within a Java project using the Java native interface (JNI) programming framework. This gives the comfort of having full access to the Android (API) with the performance advantages of the C++ language for the validation of the images. Using this Java native interface (JNI) framework the validation methods are integrated in both the applications as shown in Figure 20.

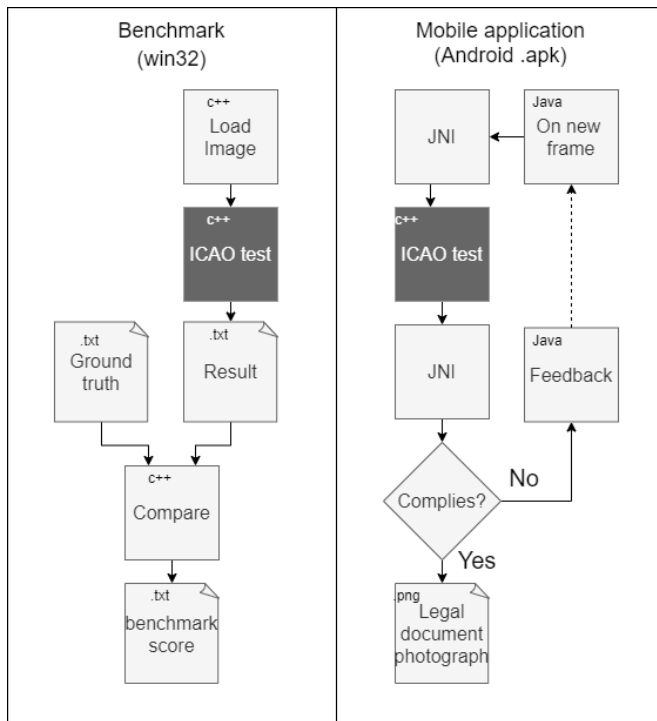


Fig. 20: Integration of the ICAO test algorithms

A. Application architecture

A simplified representation of the global application architecture is shown in figure 21. The representation only contains the core functions of the application, a more complete overview is given in the user interface section.

1) *Update peripherals*: The landmark/segmentation models are not embedded into the application to minimize the size of the Android application package and allow updating of the models without modifying the application. Instead the models are stored in the local user data directory of the smartphone. The update peripherals function verifies the version of the files and automatically downloads the new files in the event of an update.

2) *Frugally-deep*: The segmentation models and hat/cap classifier are trained with the Keras deep learning library. Since this library is written in Python the models cannot directly be

used in a C++ project. The frugally-deep library [6] offers a solution for the conversion and interpretation of Keras models in a C++ environment.

3) *ICAO test*: The creation of a segmentation mask with the semantic segmentation model takes around 0.2 seconds which is equivalent to a frame rate of 5 frames per second. To give more real-time experience to the user, the segmentation masks are created in the background task. During the creation of these segmentation masks, the application will continue showing new frames. However, these frames will not be evaluated by the ICAO test algorithms. Because of this, the application is capable of running with a frame rate of 10 frames per second, but the user feedback is delayed by one frame. (See appendix for a more extensive schematic representation of this function.)

B. User interface

The main goal of the application is aiding the user in the process of creating a legal document photograph. However, the requirements of a legal document photograph differ for each individual. For example, a child under the age of six does not have to meet to the requirements 17, 25, 26. The application should therefore offer the possibility to enable or disable the image validation of certain ICAO requirements. Some people might already have a photograph of which they assume that it complies to the ICAO requirements. In that case it might be more convenient to test the existing photographs rather than creating a new one. Based on these characteristics a simple user interface was devised which offers these functionalities to the user (see Figure 22). The rest of the section will give a brief description of the user interface shown in figure 22.

1) *Main menu*: The application home screen is a simple menu, the functionality of the buttons in the main menu is briefly explained in the rest of this section.

2) *Create a new photo*:

a) *Auto capture*: With this function enabled, the application will automatically take a pictures once all the requirements are met. After disabling this feature the user can manually capture an image with the capture button.

b) *Voice assistant*: The application has in addition to the onscreen gestures and text messages the ability to guide the user with a voice assistance. This feature is enabled by default but can be disabled by the user.

c) *Camera*: There are a lot of smartphones available and they all have different specifications. Some smartphones have multiple front face cameras and therefore it is desirable to have an option to switch between camera's.

3) *Use existing photo*: The application can also validate existing photographs. After verifying the photo the application will indicate whether the photo conforms for each ICAO requirement.

4) *Settings*: Since some of the requirements are not mandatory for users of a certain age or country the application has the ability to disable certain ICAO tests in the settings menu.

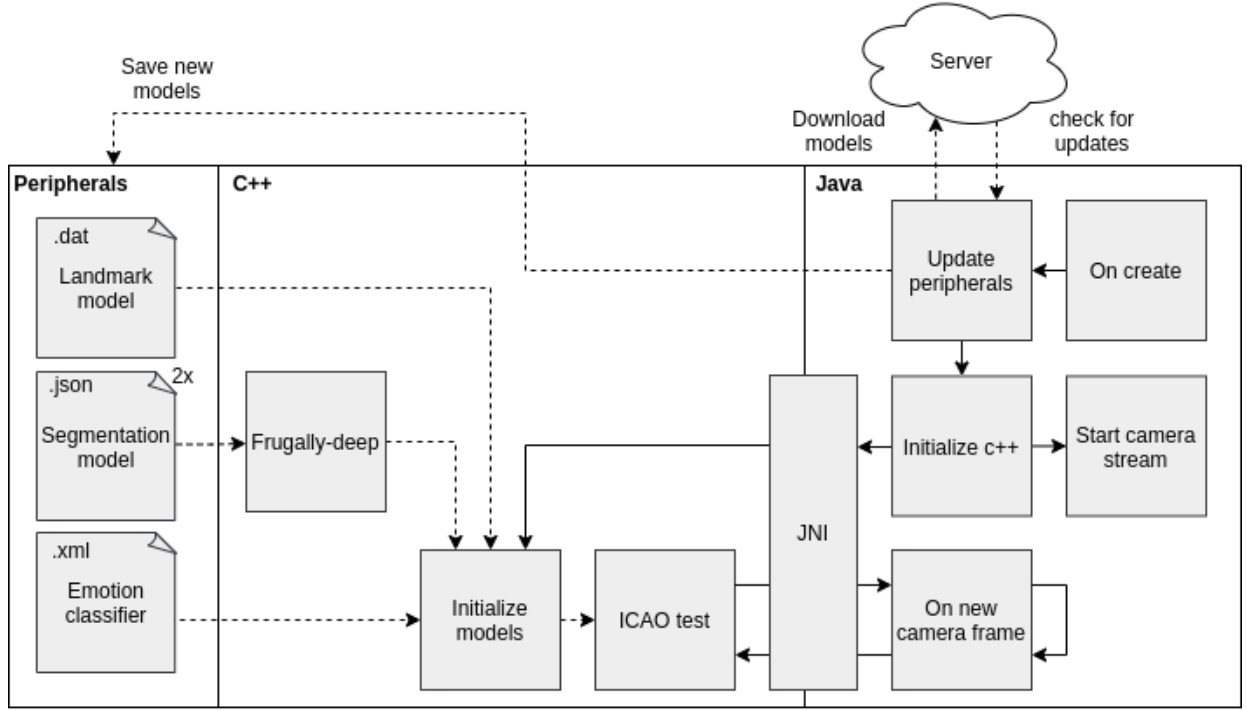


Fig. 21: Schematic representation of the application architecture

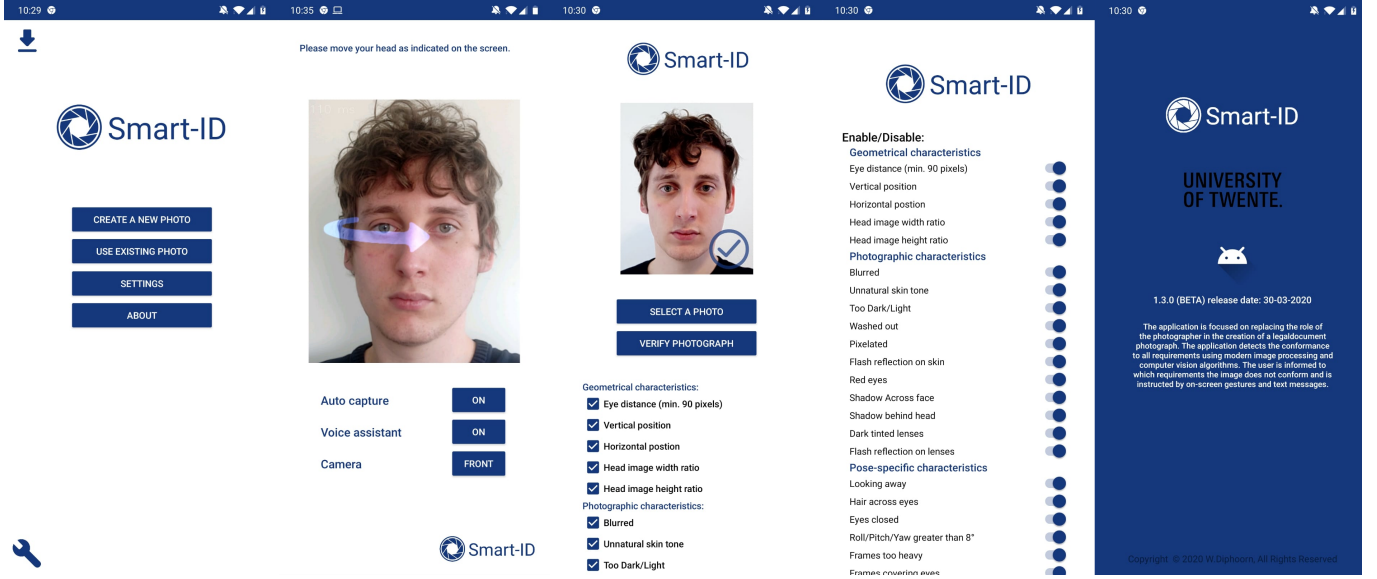


Fig. 22: user interface of the application

V. TESTING AND VALIDATION

1) *Image validation:* The work of [6] provides a benchmark tool to validate the ICAO test software. The benchmark results consist of an equal error rate (EER) and a rejection rate for each characteristic. The rejection rate is the percentage of images that could not be validated by the software development kit (SDK). For the proposed method an image will only be rejected when no face has been detected. Over the complete set of images, 1.03% has been rejected. Since the multiple faces in image detection (ICAO requirements 29) completely depends on the accuracy the face detection, the

evaluation of this test can be derived from this percentage. The benchmark results of the present paper and previous work are shown in Table III. Unfortunately, the benchmark tool provides no information on the accuracy of the geometric features. However, the accuracy of the geometric features are completely dependent on the accuracy of the facial landmarks (see Figure 4). The landmark model used in this paper has already been extensively researched in the work of [13]. Therefore, it could be concluded that the geometric features are already correctly validated.

Since this paper uses a more extensive list of requirements in

TABLE III: FICV-1 Benchmark results

ICAO test:	SDK1(*1):		SDK2(*2):		BioLabSDK(*3):		UFPB(*4):		UFCG(*5):		Proposed Method:	
	EER	Rej.	EER	Rej.	EER	Rej.	EER	Rej.	EER	Rej.	EER	Rej.
6 Blurred	26.0%	8.9%	48.1%	0.6%	5.2%	0.0%	-	-	-	-	15.1%	0.4%
7 Unnatural skin tone	18.7%	4.8%	50.0%	0.8%	4.0%	0.2%	-	-	3.7%	0.0%	1.2%	0.0%
8 Too dark/light	-	-	3.1%	0.0%	4.2%	0.0%	-	-	-	-	3.5%	0.0%
9 Washed out	-	-	40.8%	0.2%	9.6%	0.0%	-	-	-	-	0.0%	0.0%
10 Pixelation	-	-	0.0%	0.0%	1.3%	0.0%	1.7%	0.0%	-	-	0.0%	0.0%
11 Flash refl. on skin	5.0%	2.7%	50.0%	7.5%	0.6%	0.0%	-	-	1.3%	0.0%	0.8%	2.7%
12 Red eyes	5.2%	4.5%	34.2%	0.0%	7.4%	0.0%	-	-	-	-	13.4%	1.0%
13 Shadow across face	36.4%	8.1%	-	-	13.1%	0.4%	-	-	7.7%	2.5%	10.2%	0.0%
14 Shadow behind head	-	-	-	-	2.3%	0.2%	-	-	-	-	7.4%	1.9%
15 Dark tinted lenses	-	-	-	-	1.9%	0.2%	-	-	-	-	0.8%	1.3%
16 Flash refl. on lenses	-	-	-	-	2.1%	0.0%	-	-	-	-	1.4%	1.0%
17 Looking away	27.5%	7.1%	-	-	20.6%	0.0%	-	-	-	-	16.4%	2.7%
18 Hair Across eyes	50.0%	81.9%	-	-	12.8%	0.0%	11.9%	3.4%	-	-	10.3%	3.8%
19 Eyes closed	2.9%	3.1%	-	-	4.6%	0.0%	-	-	-	-	6.5%	0.0%
20 Roll/Pitch/Yaw > 8°	-	-	26.0%	2.9%	12.7%	0.2%	-	-	-	-	7.1%	1.5%
21 Frames too heavy	-	-	-	-	5.8%	0.0%	-	-	-	-	2.1%	0.9%
22 Frames covering eyes	50.0%	62.3%	-	-	6.3%	0.0%	-	-	-	-	18.7%	0.2%
23 Hat/Cap	-	-	-	-	14.0%	0.0%	-	-	-	-	7.8%	0.2%
24 Veil over face	-	-	-	-	2.5%	0.0%	1.2%	0.5%	-	-	1.7%	4.6%
25 Mouth open	3.3%	52.1%	-	-	6.2%	0.0%	4.2%	0.2%	-	-	0.8%	0.0%

*1 Benchmark results unknown commercial SDK

*1 Benchmark results unknown commercial SDK

*3 Benchmark results BioLabSDK [8]

*4 Benchmark results Vsoft & Federal University of Paraíba (UFPB) [21]

*5 Benchmark results Vsoft & Universidade Federal de Campina Grande (UFCG) [2]

comparison to the work of [8], some requirements can not be validated using the Benchmark tool. For these requirements, a new test-set has been generated from the FICV-test dataset that takes into account the associated features. The varied background / Object background detection test can not be validated with the benchmark of [8] for two reasons. First, the [8] expects a test result for an object detection and varied background detection. The algorithm however, can make no distinction between those two attributes. Secondly, as mentioned before, the image is cropped so that it suffices to the geometrical requirements of a passport image. Therefore, the algorithm cannot detect any varying regions / objects that are not within the cropped image (see Figure 4). The results can be found in Table IV.

2) *User feedback*: The application and its feedback function are validated with a small group of participants. During the experiment the participants were asked to create a photograph with the application. The participants were able to successfully create a legal document photograph. A few examples of the resulting photographs are shown in this paper due to the size limit of the paper and the privacy concerns of the participants (see Figure 23).

TABLE IV: FICV-test results

ICAO test:	EER:
26 Non-neutral expression	8.6%
27 Wrong background colour	9.6%
28 Varied background/Object in background	18.0%



Fig. 23: legal-document photographs created with the smart-phone application

VI. DISCUSSION

Legal document photographs are normally taken by a professional photographer, because these photos need to conform to all ICAO guidelines and individuals are mostly unfamiliar with those guidelines. This paper therefore proposed a set of requirements on which legal document should conform, tested the requirements and developed an application in which individuals can make their own legal document photograph. The requirements are based on previous work and also tested in comparison to those papers. Overall it can be concluded that the conformance of the tested requirements can be estimated with reasonable accuracy for most ICAO requirements. The biggest improvements on algorithms accuracy are achieved for the following requirements: the mouth open, washed out, looking away, roll/pitch/yaw $> 8^\circ$ and hat/cap classifier. The washed out and pixelation algorithms are even so much improved that they have an equal error rate (EER) of 0%.

Unfortunately, not all image validation methods improved in comparison to previous work, some EER scores decreased. Especially the requirements frames covering eyes, blurred, red-eyes and shadow behind head have a significant decrease in accuracy.

First, the accuracy drop for the red-eyes and frames covering eyes detection algorithm could be explained by the fact the detection often fails when a user wears tinted glasses. This is due to tinted glasses often resulting in a misplacement of the facial landmarks associated with the eye. If this is the case, the accuracy of this algorithm could easily be improved by validating this requirement only when the person is not wearing tinted glasses. So, instructing the user to take off their tinted glasses. The high accuracy of the tinted lenses

detection algorithm should make this solution easily possible. An alternative solution could be to use another landmark shape predictor that is less sensitive to tinted glasses.

Secondly, the results of the blurred image detection algorithm are not sufficient. This could be due to the fact that this paper used the method of [23] which is quite simplistic. A more advanced algorithm could be a solution for this, as is proposed by [8].

Next to that, the EER of the flash reflection on skin detection method in this paper was higher than the EER of the paper by [8]. Both methods estimate the compliance to the ICAO requirement based on the number of high intensity pixels in the Y channel of the YCbCr color-space in the face region. However, the method of [8] defines the skin region as a rectangular region around the face. The present paper uses the area within the facial landmarks as the skin region (see figure 2). The conception was that the algorithm used in this paper could do a more accurate evaluation of flash pixels since the squared region around the face used by [8] also included non skin pixels. Unfortunately, the method used in this paper excluded the pixels on the forehead, so they were not evaluated. Therefore, the lower accuracy could probably be explained by the flashes on the forehead that are not discernible within the defined skin region.

Although the shadow across face detection has improved in comparison to the work of [8], it did not surpass the accuracy of [2]. The color-based filtering method of [2] appears to be a more accurate method than the method proposed in this paper. Therefore, it could be concluded that the method of [2] should be used instead of the method used in this paper.

The difference between the EER score for the shadow behind head detection algorithm of [8] and the method proposed by this paper could be explained due to the usage of different background detection methods. The background detection method used by the present paper is somewhat simplistic. The result of the background detection method is merely an approximation of the background pixels rather than an accurate background segmentation mask. The results of this paper suggest that a more advanced background detection is needed in order to estimate the compliance to this requirement more accurately.

Also the closed eyes detection shows a difference in EER score among the methods used by the SDK1, BioLabSDK [8] and the present paper. The SDK1 shows the highest accuracy rate with an EER of 2.9%. However, no information is available on how this method exactly works. The second best accuracy rate is the result of BioLabSDK [8] with an EER of 4.6%, but the result is not overwhelming in comparison to the EER of this paper (6.5%). Furthermore, the implementation of the method proposed in the present paper is more straightforward whenever facial landmarks are available.

Lastly, based on the rejection rate per requirement it is evident that the face detection algorithm has difficulties with detecting occluded faces. Both veil over face and hair across eyes have a relatively high rejection rate. To decrease these rejection rates a face detection model that is less susceptible to occlusions should be used.

Looking at the results presented in Table IV there can

be observed that all the tests results within this table have a relatively high EER. Many studies addressed the issue of accurately estimating a human expression. This method was mainly chosen for its simplicity. There are however, many advanced emotion recognition techniques available that would probably achieve a higher accuracy.

The wrong background colour and varied background/object in background detection algorithm suffer from the same effect as the shadow behind head algorithm. As stated earlier, the accuracy of the background detection is quite poor, an more advanced background detection algorithm will significantly improve the accuracy of these ICAO test algorithms.

As mentioned before, the image validation algorithms were implemented in a smartphone application. The efficacy of this application was validated with a small group of participants. Most participants struggled during the experiment with finding a suitable background with the correct lighting conditions. Simplifying this process would most probably refine the user experience. Two suggestions are presented. First, the application could provide more information on the requirements. For example, a user tutorial shows the user how to create a scene with the correct lighting conditions. Secondly, a need for a neutral white/gray background could be negated by an accurate background replacement algorithm.

Another frustration for the users was the false negative feedback provided by the application. The application occasionally provides false negative feedback, due to the imperfection of the detection algorithms. This problem occurs most of the time in the red eye and looking away detection. This is argumentatively due to the EER of these detection algorithms. Improving the accuracy of these detection algorithms would significantly refine the user experience of the application. The benchmark results of SDK1 (see Table III) show that it is possible to have a more accurate red eye detection algorithm. However, the localisation of the eye pupil is a more challenging task since the EER of this detection algorithm is quite high despite the improvement made by this paper in comparison to previous work.

Despite some false negative feedback and difficulties with finding a suitable background, all participants were able to create a legal document photograph with the application.

VII. CONCLUSION

This research aims to answer the question: How to develop an application that replaces the role of the photographer in the creation of a legal document photograph? The requirements of a legal document photograph are listed in section II-A of this report. Hereby the question on the requirements of a legal document photograph has been answered.

The paper provides a method for testing an image on the conformance to every requirement. The accuracy of most of these methods have been validated using the FICV-ongoing benchmark. Based on the results it can be concluded that the application can estimate the conformance for the most of the requirements with an acceptable accuracy. While there is still room left for improvement, the accuracy of the majority of the algorithms are improved in comparison to earlier work.

Hereby the second sub-question on how to test a photograph on the conformance to these requirements has been answered.

A user feedback method is provided for every requirement of a legal document photograph. The ICAO test algorithms and feedback mechanisms are implemented in a mobile application. This application is capable of providing feedback for every ICAO requirement in a real-time user experience.

The experiment with real users shows that it indeed possible to create a legal document photograph with mobile application even without knowledge of the ICAO requirements. So, the application offers an alternative to a professional photographer. There are, however, still some ICAO requirements that the application cannot validate accurately. Therefore, the professional photographer is still the more reliable option. Further improvements are necessary to achieve the reliability of a professional photographer.

Future research should focus on improving the accuracy of the validation of ICAO requirements. This would not only improve the quality and reliability of the photographs, but it will also increase user experience. Besides, it would be interesting to test the application in a larger experiment with more users. Preferable in a setting with a professional photographer as well. In that case, the photographs of the application could be compared with the photographs of the professional photographer in order to check reliability of both photographing ways.

REFERENCES

- [1] U. Ahlvers, R. Rajagopalan, and U. Zölzer. Model-free face detection and head tracking with morphological hole mapping. In *2005 13th European Signal Processing Conference*, pages 1–4, 2005.
- [2] I. L. P. Andrezza, E. V. C. L. Borges, R. A. T. Mota, and J. J. B. Primo. Facial compliance for travel documents. In *2016 29th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 166–172, 2016.
- [3] Lingyun Wu Ping Luo Cheng-Han Lee, Ziwei Liu. Maskgan: Towards diverse and interactive facial image manipulation. *CoRR*, abs/1907.11922:1–20, 2019.
- [4] Nouredine Cherabiti. Circular hough transform for iris localization. *Computer Science and Technology*, 2:114–121, 2012.
- [5] Marco Couto, Francisco Ribeiro, Rui Rua, Jácome Cunha, João Fernandes, and João Saraiva. Energy efficiency across programming languages: how do energy, time, and memory relate? In *SLE 2017*, pages 256–267, 2017.
- [6] Dobiasd. Github [online], available: <https://github.com/dobiasd/frugally-deep>. 2018.
- [7] Bernadette Dorizzi, Raffaele Cappelli, Matteo Ferrara, Dario Maio, Davide Maltoni, Nesma Houmani, Sonia Garcia-Salicetti, and Aurélien Mayoue. Fingerprint and on-line signature verification competitions at icb 2009. *Advances in Biometrics : Third International Conference, ICB 2009, Alghero, Italy, June 2-5, 2009*, 5558:725–732, 06 2009.
- [8] M. Ferrara, A. Franco, D. Maio, and D. Maltoni. Face image conformance to iso/icao standards in machine readable travel documents. *IEEE Transactions on Information Forensics and Security*, 7:1204–1213, 2012.
- [9] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *CoRR*, abs/1704.04861:1–9, 2017.
- [10] R. V. Hsu, J. Shah, and B. Martin. Quality assessment of facial images. In *2006 Biometrics Symposium: Special Session on Research at the Biometric Consortium Conference*, pages 1–6, 2006.
- [11] A. Kae, K. Sohn, H. Lee, and E. Learned-Miller. Augmenting CRFs with boltzmann machine shape priors for image labeling. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2019–2026, 2013.

- [12] V. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1867–1874, 2014.
- [13] Philipp Kopp, Derek Bradley, Thabo Beeler, and Markus Gross. *Analysis and Improvement of Facial Landmark Detection*. University of British Colombia, Disney Research, ETH Zurich, Computer Graphics Laboratory, 2019.
- [14] Askhay C Lagandula. Towards data science [online] mouse cursor control using facial movements - an hci application, available: <https://towardsdatascience.com/mouse-control-facial-movements-hci-app-c16b0494a971>. *Towards Data Science*, 2018.
- [15] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, pages 3730–3738, 2015.
- [16] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pages 94–101, 2010.
- [17] Satya Mallick. Learn opencv [online] head pose estimation using opencv and dlib, available: <https://www.learnopencv.com/head-pose-estimation-using-opencv-and-dlib/>. 2016.
- [18] A. Martinez and Robert Benavente. The ar face database. *Tech. Rep. 24 CVC Technical Report*, 24:1–11, 1998.
- [19] Dario Maio Matteo Ferrara, Annalisa Franco. A multi-classifier approach to face image segmentation for travel documents. *Expert Systems with Applications*, 39/9:8452–8466, 2012.
- [20] Nuwan Munasinghe. Facial expression recognition using facial landmarks and random forest classifier. In *17th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2018)*, pages 423–427, 2018.
- [21] R. L. Parente, L. V. Batista, I. L. P. Andrezza, E. V. C. L. Borges, and R. A. T. Mota. Assessing facial image accordance to iso/icao requirements. In *2016 29th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 180–187, 2016.
- [22] Adam Paszke, Abhishek Chaurasia, Sangpil Kim, and Eugenio Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *CoRR*, abs/1606.02147:1–10, 2016.
- [23] J. L. Pech-Pacheco, G. Cristobal, J. Chamorro-Martinez, and J. Fernandez-Valdivia. Diatom autofocusing in brightfield microscopy: a comparative study. In *Proceedings 15th International Conference on Pattern Recognition ICPR-2000*, pages 314–317 vol.3, 2000.
- [24] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4510–4520, 2018.
- [25] Giuseppe Messina Giovanni Puglisi Sebastiano Battiato, Arcangelo Ranieri Bruna. *Image Processing for Embedded Devices*. Bentham Science, Sharjah, united arab emirates, 2010.
- [26] Satoshi Suzuki and Keiichi Abe. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 30:32–46, 1985.
- [27] Yingli Tian Takoe Kanade, Jeffrey F.Cohn. Comprehensive database for facial expression analysis. In *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, pages 46–53, 2000.
- [28] Fabian Timm and Erhardt Barth. Accurate eye centre localisation by means of gradients. In *VISAPP 2011 - Proceedings of the International Conference on Computer Vision Theory and Application*, pages 125–130, 2011.
- [29] ISO/IEC JTC1 SC17 WG3. Portrait quality (reference facial images for mrted). Standard, International Organization for Standardization, Geneva, CH, April 2018.

APPENDIX A
SCHEMATIC REPRESENTATION OF THE C++ ICAO TEST FUNCTION

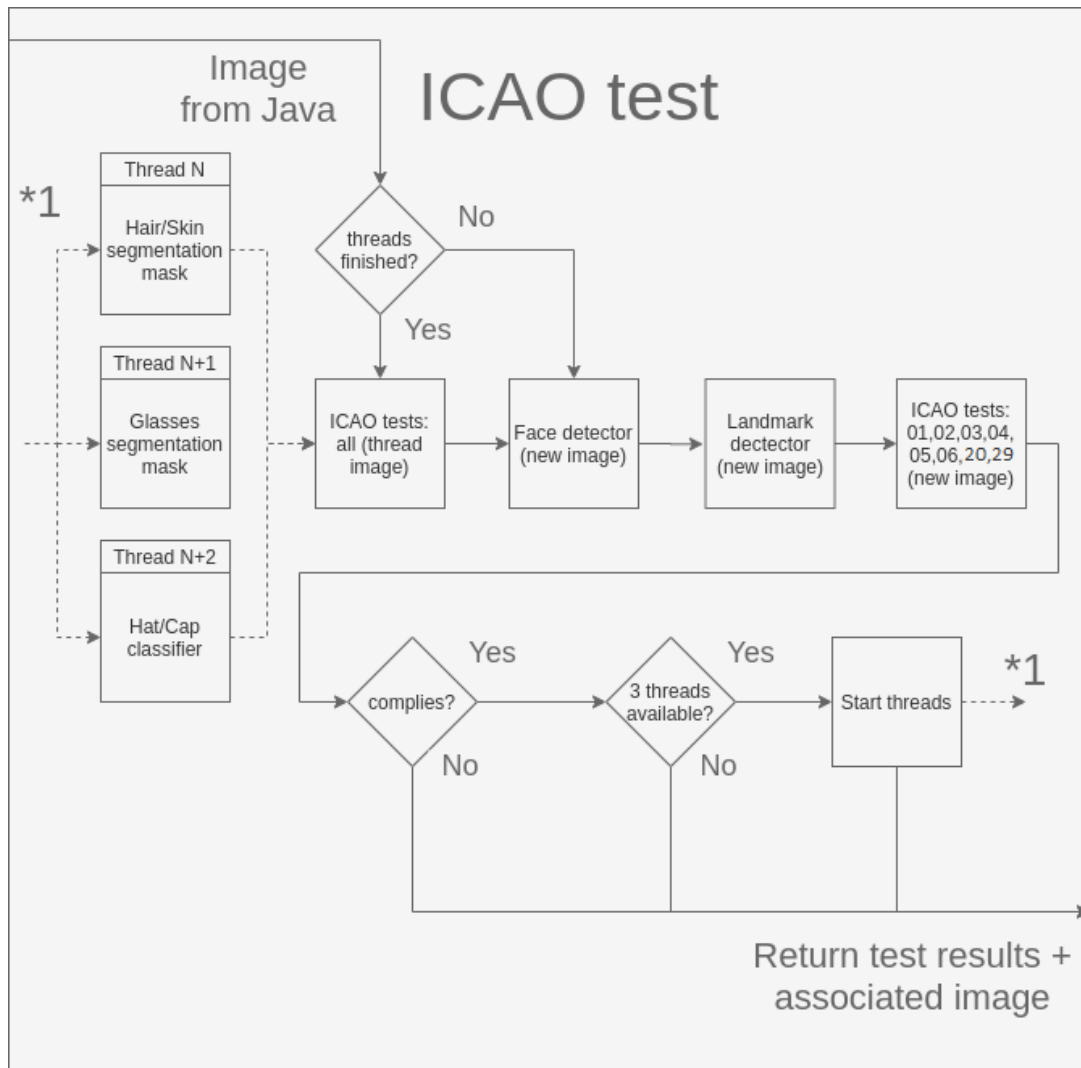


Fig. 24: Schematic representation of the C++ ICAO test function