



**HAL**  
open science

## Instant Rephotography

Juliano Franz, Anderson Maciel, Luciana Nedel

► **To cite this version:**

Juliano Franz, Anderson Maciel, Luciana Nedel. Instant Rephotography. 17th IFIP Conference on Human-Computer Interaction (INTERACT), Sep 2019, Paphos, Cyprus. pp.208-227, 10.1007/978-3-030-29387-1\_12 . hal-02553905

**HAL Id: hal-02553905**

**<https://inria.hal.science/hal-02553905v1>**

Submitted on 24 Apr 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# Instant Rephotography

Juliano Franz, Anderson Maciel, and Luciana Nedel

Institute of Informatics, Federal University of Rio Grande do Sul (UFRGS), Porto Alegre, Brazil {jmfranz, amaciel, nedel}@inf.ufrgs.br

**Abstract.** When traveling, a very common issue is to ask for the help of a stranger to take a photo on a touristic spot. Indeed, often the results of this short collaboration are not as previously imagined. In this paper, we introduce the concept of *instant rephotography*, a technique based on a simple and intuitive user interface, and in an efficient and low cost algorithm that finds the matching between two images. The user interface guides the photographer to position the camera in the right place that was once imagined by the camera owner, automatically taking the new photo. To correctly choose the new photo, we propose an algorithm based on ORB (Oriented FAST and Rotated BRIEF) feature points acquired from the reference and current photos, and on inertial sensors embedded in the photo device, a smartphone. To validate the concepts introduced, we implemented an Android App called Smart-Tourist Camera, conceived to help tourists to take good pictures during travels. Three different user studies involving 124 volunteers were conducted and results have shown that the photographs taken with the application are perceived as better than otherwise, and that users would use the App during their travels.

**Keywords:** Computer Photography · Sensors Fusion · Rephotography

## 1 Introduction

Rephotography is the act of recreating a photograph in the same site with a time span between the two images [2]. Images created in that matter are usually taken from the same point of view and with the same framing as a reference image made in the past. Examples include a series of “Then & Now” books that show two views of the same sites in a big city around a century apart [13] and are being used for historical purposes.

In this paper, we contribute in computational photography with the introduction of a novel rephotography approach. We motivate our work around the travel photography problem, a scenario where a typical tourist needs help from a stranger to take a picture of him/herself. Using the conventional approach, the outcome is rarely as good as expected. There are several reasons for that. Most people are in a hurry and, when someone asks them for this favor, they often try to do it as quickly as possible. Others have time but might not have the skills to correctly frame a photo. Those that know what they are doing are uncommon, and still, that does not mean the result will be satisfying because

people just think differently. Furthermore, even the ones that find a stranger with time and skill to help them, might not be able to explain what he or she wants due to language barriers. In this context, however, our assumption is that if the vantage point, lens coverage and composition could be fixed in some way, all the stranger would have to do is to press the shutter button.

More specifically the approach proposed in this work focus on trying to solve a common scenario in travel photography: *a person (tourist) wants its photo taken at a touristic place according to a specific framing, i.e. camera’s yaw angle, point of origin and orientation, however, s/he needs to rely on another person (stranger) to operate the camera thus relinquishing control over the camera parameters.* STC focus on bringing back the control over the framing of the camera back to the camera owner by allowing s/he to set a frame of reference (or picture of reference) which will be used by the application to guide the stranger before capturing the photograph. STC enables the tourist to pre-frame the picture that s/he wants to be part of without the need to explain it to the stranger. By doing this, it minimizes, if not discards, external interference on how the image should look. STC also makes it easier for the stranger to participate in the transaction because s/he does not need to think about picture quality as all the stranger has to do is follow simple on-screen instructions of translation and rotation which are guiding the camera towards the pre-set reference

Having in mind that rephotography techniques are used to illustrate such “then and now” situations, in this paper we introduce the concept of *instant rephotography*, and present a fast method to help users to recreate an image made just a few seconds earlier. Differently from the common uses of rephotography, in this case, the focus application is casual and the techniques used should provide a transparent and easy to use user interface. This implies, as well, in providing a very fast algorithm to detect the correct camera position and orientation. Figure 1 illustrates the ideas behind *instant rephotography*.

The main contributions of this work are:

- An efficient and power-friendly algorithm to measure the distance between two photos and to guide the photographer to correctly position the camera
- A clear and intuitive user interface that can be operated by anyone without any instruction
- The Smart-Tourist Camera (STC), an Android App that allows instant rephotography using a smartphone

The remainder of this paper is organized as follows. The next section presents previous related work. Section 3 introduces the instant rephotography technique and gives details about its implementation. Section 4 describes the three user experiments we conducted to verify the system effectiveness and acceptability, and discusses the results achieved. Finally, we discuss our findings and implications in Section 5 and Section 6 concludes this work and presents directions for future work.



**Fig. 1.** The tourist makes a photo with the tree (a) exactly as she wants. Then, she hands the camera to somebody else, asking the other person to take a picture of himself in the same landscape. The algorithm detects keypoints (red circles) in the original picture (a); the user interface guides the photographer to place and point the camera in the suitable position and orientation; and a new photo is automatically taken when the desired pose is reached (b).

## 2 Related Work and Background

Authors have discussed that rephotography techniques are important due to the fact that they can be used to study the evolution of locations, buildings and architectural proprieties over time. However, as discussed by Lee et al.[10], recreating a photograph is a challenging task because in order to recreate a viewpoint one needs to fix six degrees of freedom. Lee et al. propose that, because of such challenge, rephotography should be done off-site in post processing. In their work they combine a series of new images from different angles with a 3D point cloud and depth map of the location. The data is then paired with the reference photograph in order to recreate the photograph by combining the new images to be as similar as possible to the reference.

Shingu et al. [19] propose another technique to help users recreating a previous viewpoint. In their work, they make use of an augmented reality system to properly repeat a photography. ARToolkit markers are placed in the scene to extract information of the camera position and orientation in the real world. With that information they render a visual marker on the camera screen indicating to the user where he should place the camera in order to recreate the previous viewpoint.

Real time rephotography algorithms have also been explored before [2]. Bae et al. introduced a solution to recreate photographs from the past using a reference photograph and image processing techniques *in loco*. In their work, they have shown that it is not possible to recreate a photograph relying only on naive composition, i.e. trying to recompose an image just by looking at the old one. According to them, there is a need for computational assistance. Their approach, however, required a time costly calibration phase, as well as the use of a computer attached to the camera to handle the image processing. The entire



process limits the user freedom and requires around thirty minutes to recreate a single photograph which is unacceptable in many situations, such as the travel photography scenario.

Similar techniques that are used for rephotography were used by Vazquez and Steinfeld [20] to help vision impaired persons to take better street photographs to document issues in the city. They suggest a technique for smartphones that tries to help the user to fix camera roll and extracts region of interest areas from the image, in real time, helping the user to move the camera to keep them properly framed. On the same topic, Balata et al. [3] also proposed a smartphone based guidance tool to assist vision impaired people to take better composed photos. In their work, they introduce two photographic guiding techniques (central and golden-ratio) and two methods for guiding users: voice and vibration. They found no statistical difference between the performance of blind folded participants however self-reported comfort with voice guide was higher than with vibration.

Wang et al. [21] proposes a technique analogous to ours: a smart camera application used to suggest where the tourist should stay in the picture for a better composed final image. They use image segmentation and photographic rules to analyse a reference photo. The result from segmentation is used to render a shadow in the camera live-view which then indicates where is the ideal standing position for the person who wants to appear in the photo to stand at. The photographer uses the shadow to guide the subject to its location. Rawat et al. [16] also discuss a smart camera application to aid users in taking better composed using machine learning feedback. They use a collection of geo-tagged photos from public sources to train the guidance algorithm. Users of their method can then use their assistance method based on their current location in the world. The system calculates a score regarding the aesthetics of the image and guides the user to maximize the score.

HCI researchers also explored how camera applications in modern smartphones can create the opportunity for collaborative photo-taking. Jarusriboonchai et al. [8] designed and evaluated a camera application that is shared across two mobile devices. They explore two collaborative ways of taking photographs: first using one device and sharing it between two participants and second, using two devices with different functions one as the trigger and the other one as the viewfinder. They found that both techniques increased collaboration when compared to a baseline with no sharing (two separate independent devices) while the second method proved to further increase interaction between participants. James and Ünlüer [22] also explored collaborative photography but in a manner closer to what our work proposes: bringing strangers together to create an image. One stranger is the subject—or the person who ask for help—and the other one is the photographer—or the person who is willing to help. They developed an application that can notify nearby users with the app installed, using embedded geo-location, that someone needs help to have their photo taken.

One challenge that researchers had until a few years ago to properly develop real time rephotography solutions was the fact that was almost impossible to deploy algorithms into cameras available in the market. This problem was broadly

discussed by Levoy [11]. With a smartphone’s camera and processor improvement in the past few years it became easier to test new techniques outside laboratory environment. An alternative to smartphones is the Frankencamera proposed by Adams et al. [1] which can be assembled using off-the-shelf components found in electronics stores.

The implementation of instant rephotography requires the identification of feature points and then, the re-orientation of the camera to match the desired frame. While the identification of feature points is based on computer vision algorithms, the re-orientation of the camera can be achieved by tracking the movements of the camera.

Feature detectors and extractors are used since the beginning of computer vision. In the context of rephotography, detectors can be used to analyze a reference photo and to detect features to be used as cues to frame a new image. Scale-Invariant Feature Transform (SIFT) [12] is an algorithm to detect and describe local features in images which is largely used mainly because it is capable of finding keypoints that are invariant to location, scale and rotation. Mikolajczyk and Schmid [14] tested SIFT against other descriptor methods including the author’s method. SIFT outperformed most of the algorithms tested.

Another common feature descriptor is the Speeded Up Robust Features (SURF) [4], which is partially based on SIFT. SURF is several times faster and more accurate than SIFT. However, both SIFT and SURF are not intended for use in mobile applications because of their high computational cost [18, 24].

A better algorithm for mobile devices is the Oriented FAST and Rotated BRIEF (ORB) introduced by Rublee et al. [18]. ORB is a fusion of FAST (Features from Accelerated Segment Test) keypoint detector [17] and BRIEF (Binary Robust Independent Elementary Features) descriptor [5] with many modifications to enhance the performance, and is an interesting alternative for both SIFT and SURF. Because ORB is more adapted to the limited capability of mobile devices it was chosen as the base algorithm for our work.

### 3 Smart-Tourist Camera

Our approach for instantaneous rephotography can be better understood when described in the scenario of tourist photography. In this section we illustrate our technique and its materialization into the Smart-Tourist Camera (STC) application for Android smartphones. As mentioned before, STC aims to solve the problem that occurs when someone needs the assistance of a third party to take photographs for them, a very common situation when the tourist wants to be part of the photography (see Figure 1). Nevertheless, the approach can be generalized for a diversity of applications, such as crime-scene reconstruction, augmented reality systems, analysis of construction sites, etc. The software was designed based on three guiding principles:

1. It has to be simple to use
2. It should have an easy to learn interface
3. It has to complete the task in a short period of time

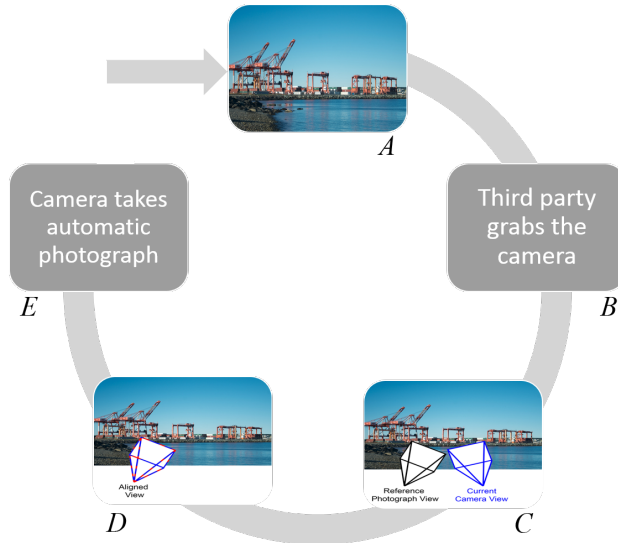
### 3.1 System Overview

The STC main objective is to avoid the need for the stranger to think on how the photography should be framed thus minimizing mistakes. By doing so the software take most of the responsibility away from the stranger.

STC consists of an application that can be deployed on any smartphone or even embedded in modern compact cameras. The minimum requirement is that they have an inertial measurement unit (IMU) with at least 6 degrees of freedom (DOF) and computational power to run computer vision algorithms. The prototype developed in this work was implemented on an Android mobile phone.

The software works by using a reference photo and a frame of reference that was set by the camera owner. Once the owner sets the reference frame, the phone can be delivered to anyone in the crowd and all this person has to do is to follow the on-screen instructions.

The application works as follows: first the tourist goes where s/he wants the camera to be, and takes a reference photo from the scene. Then, s/he delivers the camera to the third-party, here called a user. The user then points the camera towards the scene and follows the on-screen instructions. Once the camera detects that it is close enough to the reference photo it automatically takes a series of photos. The application cycle is represented in Figure 2.



**Fig. 2.** Application pipeline: User takes a reference photograph (A); stranger assumes control of the application (B); stranger follows the on-screen instructions (C); application detects that the current view is similar to the reference photo (D); instant rephotography is created (E).

### 3.2 User Interface

A successful, and thus fast, interaction with the application is only possible if the UI is simple enough to be used by someone with no previous training. This work differs from the one proposed by Bae et al. [2] specially on the interface. Their work presented a complex interface that required time to use and learn, while our own has a simple interface that requires no more than 30 seconds to learn.

The user interface was designed to be as simple as possible and is divided in two different screens: the owner’s and the stranger’s interface.

The owner’s interface is used to set the reference photo and has only two interface elements: the current camera view, and a button to take the photo, as regularly seen in other smartphone camera apps (see Figure 3a). To set the reference image, the owner has only to point the lens to the desired target and hit the button. This interaction is the same of any point-and-shoot camera.

The stranger’s interface (see Figure 3b,c,d) is more critical. It has to be easy enough for someone to understand it within a few seconds. Have in mind that probably most users will use this interface only once thus, it should be simple. The interface has no buttons, but gives two kinds of instructions for the user: how much and in which sense the camera should be rotated and translated to capture the correct photo.

The interface minimizes external distraction – i.e. any details that could distract the user coming from the camera’s image are avoided – by adding a strong vignette to the current camera frame. In doing so, the application tries to help the user to focus only on the on-screen instructions.

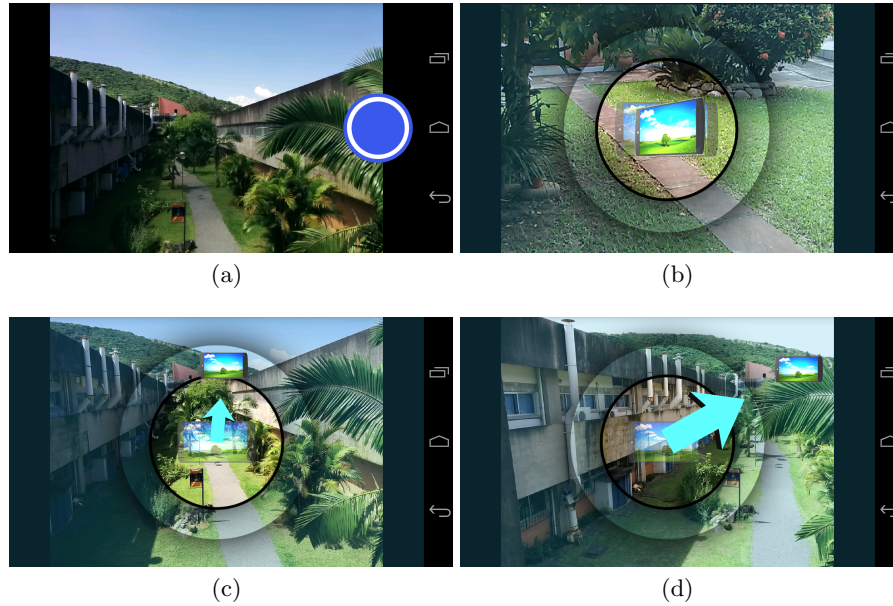
The first step in reconstructing the reference photo is to correct the camera rotation. This is done by changing the camera’s roll, pitch and yaw to set it according to the reference model. At every new frame, the camera orientation is updated and the image of the camera status is rendered on the center of the screen. All the user has to do is to align that representation with a ghost camera that shows where the real camera needs to be.

Regarding translation, the user has to move the camera towards the direction that the arrow points, placing the solid camera in the center of the screen. To help the users on how much they have to translate the camera, the arrow changes its size according to the distance. As soon as the stranger places the camera in the middle, all the screen is dimmed black, a message asking for the user to hold the camera steady is shown, a new photo is taken and the interaction ends.

### 3.3 Pose Estimation

In order to achieve the desired result, it is necessary to know where the camera is pointing at and compare it to where it should be.

Inertial measurement units (IMU) today are tiny and almost every wearable electronic or mobile device, such as cellphones and cameras, have them embedded. They provide at least 6 degrees of freedom (DOF), but most of the devices today are designed with 9 DOF units (3-axis accelerometer, 3-axis gyroscope,



**Fig. 3.** UI used for instant rephotography. Owner’s UI used to set the reference photo (a). Stranger UI: stranger rotates and translates the device to superimpose the two cameras in the interface (b). The solid camera orientation suggests the amount of rotation needed, while the blue arrow indicates the movement direction (c). The size of the arrow changes according to the amount of translation required (d).

and 3-axis magnetometer). Woodman [23] goes through the different types of devices, explaining how they can be used together to track movements.

This combined use of sensors is known as sensor fusion and is widely used in mobile devices. Sensor fusion is mostly used to track head movements in modern head-mounted displays as used by Ercan and Erdem [6] and in some cases it is used to help estimating camera position in the real world. Current state of the art IMU units have their fusion algorithms embedded in a dedicated hardware thus freeing CPU usage for that task.

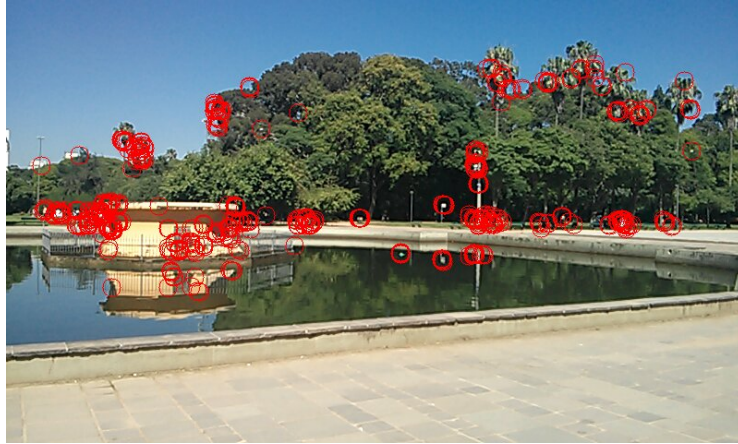
In our instant rephotography technique, sensor fusion is used to estimate the current camera orientation relative to a reference frame. This connection between both has already been used in several applications involving mobile devices, augmented and virtual reality environments [7].

In our technique, both rotation and translation axis ought to be evaluated to replicate the reference photograph properly. In the following sections we discuss how our technique handles translation and rotation during the rephotography process.

**Rotation Sensing** With a sensor fusion technique that uses all of the available IMU sensors (accelerometer, gyroscope, and magnetometer), it is possible to

obtain the quaternion (a 4 dimension representation of a rotation which does not suffer from gimbal locks) data that represents the camera orientation for every frame. Using that information it is possible to calculate the difference from the reference rotation to the current one. This rotation difference is calculated by isolating each axis and subtracting the reference angle from the current one. If every axis difference is less than a set threshold, the phone orientation is aligned with the reference frame.

**Translation Identification** To identify the camera translation, we are using image processing algorithms. At the moment that the camera takes the reference photograph, the library extracts keypoints from the image, as seen in Figure 4. After the keypoints are extracted, they are saved with the reference frame obtained from the rotation sensors. We use the keypoints from the reference image and the ones extracted from the current frame on the camera to create a distance vector, in screen space, which will guide the user towards the original camera position.



**Fig. 4.** Image showing the detected keypoints from a reference photograph.

The UI uses this distance vector to render the information for the user (the blue arrow) pointing out to where he or she should move the camera. When the magnitude of the distance vector is smaller than a defined threshold, then the correspondence between them in the rephotography process is made.

During the trials, the average number of extracted keypoints was 83 while the average of good matches was 42. This difference occurs because not every keypoint found in the reference image is present in the final image. For instance, if a moving object (a person walking) was captured when the reference image was created.

### 3.4 Shake Avoidance

Sometimes the image produced by cameras, especially in low light situations, suffers from blur originated from camera shaking. This problem can also affect our algorithm because the photographer is always moving the camera towards the correct placement. In order to reduce or avoid camera blur, once the camera is in the correct position, the application takes a sequence of photographs instead of a single one.

The intent to capture more than one image is an attempt to minimize the occurrence of image blur. However, we noticed that blur can still occur.

### 3.5 Implementation Details

The application was partially implemented in Java (Android API, and UI) and in C++ (Computer vision algorithms) using JNI as a gate between programming languages. OpenCV was used as the image processing core.

The chosen feature extractor and descriptor is the ORB which is currently the state of the art for this type of task in mobile devices. Aside from having a good resistance to noise on the images, it is also processor friendly. ORB also runs faster than traditional extractors on mobile devices [18].

The keypoints are matched using an exhaustive algorithm that takes the descriptor of one feature in the reference image and matches with all other features in the current image. The match with the smallest distance is then chosen. From each individual match we calculate a vector from the current keypoint to the reference one and extracted its angle and magnitude. Once all the individual vectors are processed, the final distance vector is created. Its magnitude (Equation 1) and angle (Equation 2) are calculated by averaging all the individual vectors from each match.

$$\|DistVector\| = \frac{\sum_{n=1}^{GoodMatches} \|n\|}{GoodMatches} \quad (1)$$

$$\Theta_{DistVector} = \frac{\sum_{n=1}^{GoodMatches} \arctan(\Theta_{GoodMatches})}{GoodMatches} \quad (2)$$

A wrapper around the Android Sensor Manager API was created to implement the Sensor Event Listener calls. The API provides a special virtual sensor called Rotation Vector. This sensor is a combination of all available motion sensors present on the phone and it represents the angular orientation of the device around an axis (x, y, or z). The refresh value of the sensors was preset to the default value of 1,000  $\mu s$ .

The main advantage of using the Rotation Vector present in the Android API is that it will always try to use the best fusion method in different mobile devices. In Algorithm 1 we present the implementation of the algorithm in pseudo-code.

---

**Algorithm 1:** STC pseudo-code

---

```

takeReferencePhotograph(referenceImage, referenceFrame);
extractKeypoints(referenceImage,referenceKeypoints);
while newFrame do
  fixRotation(referenceFrame);
  if rotationIsCorrect then
    extractKeypoints(newFrame, currentKeypoints);
    matchKeypoints(referenceKeypoints, currentKeypoints);
    calculateDistanceVector();
    if vectorMagnitude < threshold then
      takeFinalPhotograph(newFrame)

```

---

## 4 Experimental Evaluation

In order to assess our instant rephotography technique using the Smart-Tourist Camera application we proposed three experimental studies. The objective of the first study was to test the effectiveness of the technique and the usability of the interface, without any commitment with the quality of the photos taken. In the second user study, we evaluated the instant rephotography technique in a simulated travel situation. Participants asked a stranger to take a photo using a conventional camera application and then rated the photo qualitatively. The goal of this user study was to evaluate how satisfied the user is with a photo taken by someone else. Finally, in the third user study, we compared the quality of photos taken using STC with that of conventional photos taken without the help of our instant rephotography technique. The goal of this third user study was to verify if STC can improve the perceived quality of tourist photos.

We conducted the three studies in three different locations in the city of *Milwauys, End of the Universe*. We did not recycled participants from one study in the others to ensure that there was no training effect with the application. There was no compensation for participating in our study.

The proposed studies are based on probability sampling, where the target population are men and women from the age of 15 to 55 years old. The participants were selected through stratified sampling in parks around town and at the University campus.

### 4.1 Study A: User Interface Evaluation

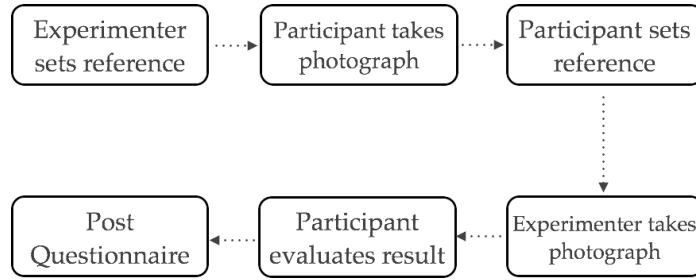
In this first study the objective was to test if the technique and the application worked properly, disregarding the quality of the final image. To be considered functional we assumed that it would have to fulfil three requirements: it should be easy, fast, and the final photo should be similar to the reference one disregarding the composition or quality of the reference image.

To assess how easy and fast it is to use the proposed interface, we evaluated if participants were able to use the application in the role of the stranger and



registered the time span between the instant they started interacting and the moment the photograph was taken.

To assess similarity with the reference photo, participants were invited to imagine and take a reference photograph where the only constraint was that it had to be in landscape orientation (to minimize the number of variables). Then, the final and the reference photos were shown and the participant was asked to answer, in a 5-points Likert scale, how similar they judge the final photo is compared with the reference one.



**Fig. 5.** Diagram demonstrating the protocol used in the user study A.

During the trials, participants first received the camera with a reference preset by the experimenter and were only asked to follow the on-screen instructions without any further instructions. Later the participant was invited to set him/herself a new arbitrary reference image – playing the role of the tourist – and the experimenter followed the instructions. Then, they were informed about how the application worked and what was its purpose. Finally, the participant was invited to answer the post questionnaire form. Figure 5 illustrates the protocol used in this study.



**Fig. 6.** Example of a reference photograph (red circles highlight the extracted keypoints)(a), and a rephotography automatically taken by a participant using STC (b).

For the user study A, we had 40 participants (55% women and 45% men) with a mean age of 26.1 and  $\sigma$  +/- 7.86 years.

**Results.** Participants averaged 8.62 seconds ( $\sigma = 4.79$ ) before STC took the automatic picture. All the users from study A were able to complete the study.

In the post task questionnaire, 90% of the participants answered that the reference and the final photographs were similar (ratings 4 and 5 of the scale), while 87.5% said that they would use the application (ratings 4 and 5) during one of their travels if the application was available. Figure 6 presents an example of a result from study A.

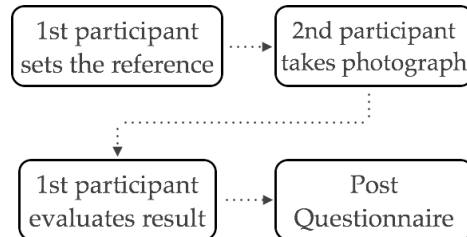
#### 4.2 Study B: Rephotography Similarity

For this second study, different participants were selected and introduced to the following scenario: *You are a tourist visiting this of Milliways. Lets assume that you wish a photo of yourself here at this park. You are going to need the assistance from a stranger to shoot it for you. Assume that this person (another participant) is that stranger. First imagine the picture without you on it and take this photo as a reference. Then, go to the place where you imagined you would be and the stranger will try to replicate the reference photograph using the application.*

After the image was taken, the first participant was asked to evaluate – using a 5-point Likert scale – how similar the final photograph was to the one they have previously done. Figure 7 illustrates the protocol for this user study.

The primary goal of this study was to evaluate our instant rephotography technique in a scenario similar to a travel situation. We achieved that by pairing participants with people that they did not know, hence strangers. The only information that was given to them was that they should follow the on-screen instructions.

In this study, we collected 40 participants (45% women and 55% men), with an average age of 29.05 and  $\sigma$  +/- 9.54 years.

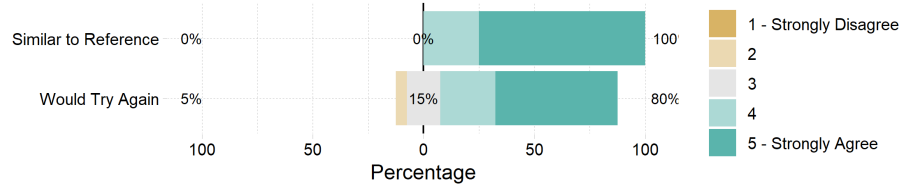


**Fig. 7.** Diagram demonstrating the protocol used for the user study B.

**Results.** The results of this study are similar to the ones from study A. The major difference between the two studies was the use of two participants in this one – in the first user study the reference photo was taken by the experimenter

–, and the fact that now the first participant must to appear in the automatic picture taken by the application.

All participants answered that the final photograph was similar to the one they had imagined (ratings 4 and 5), while 80% of the participants also answered that they would use the application during their travels if it was available (Figure 8). Figure 9 depicts a pair of images taken during this study.



**Fig. 8.** Likert responses from the second study.

### 4.3 Study C: To use STC or not to use it?

We selected a new set of participants for the last user study. The scenario was similar to the one used in the user study B. However, this time the stranger was first asked by the tourist subject to take a picture of him/her, without any assistance from the software, similarly to how it works when we travel and ask for a stranger to take a photo of us. Later, he/she was invited to take the picture again, this time using the Smart-Tourist Camera with a reference image set by the *tourist* participant. Once the stranger completed the task, the two resulting images, manual and assisted, were scrambled and shown in the smartphone screen. Then, the experimenter asked the tourist participant to choose the image that they thought was the best. Figure 10 illustrates the protocol used for this study.

Later, after all the participants finished the trials, we thought it could be interesting to have a third opinion. All the images were then combined in pairs (without and with assistance) and were randomly shown to a third set of evaluators. The evaluators were unaware of the existence of the application. For each person in this last group, a sequence of image pairs was displayed and they were asked to select the best image on each pair.

We had 20 participants (75% women and 25% men) with an average age of 32.05 and  $\sigma$  +/- 10.92 years for the first part of the experiment, and 24 participants (17% women and 83% men) with an average age of 29.29 and  $\sigma$  +/- 5.64 years for the second part.

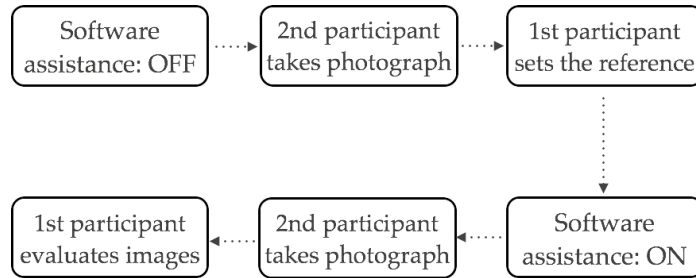
**Results** Twenty volunteers participated in the first part of this study, 10 as tourists and 10 as photographers. In the blind test, 9 out of the 10 tourists



**Fig. 9.** Reference (a) and final (b) photos for one participant of study B.

participants indicated they preferred the image taken using the rephotography assistance.

For the second part of the study, 24 people volunteered to select the image they prefer in each of the 10 pairs of photographs. Remember that those people did not know about the existence of the instant rephotography technique and only analyzed the images. Results are: 146 selections (60.8%) indicated preference for the assisted photographs, 94 (39.2%) preferred the manual photos. Analyzing from the perspective of the 10 pairs of photos, only 2 pairs received more selections for the photograph taken manually than the automatic counterpart. These 2 were exactly those that presented low focus or motion blur issues on the automatic photo.



**Fig. 10.** Diagram demonstrating the protocol of the user study C.

## 5 Discussion

Findings from the three users studies indicate that our instant rephotography technique was successful in assisting our participants to take (or re-take) similar

photos with very little effort and time. We believe that all the three initial design conditions— simple to use, easy to learn, fast interaction— were met as we observed no difficulties for the participants to follow the on screen instructions while keeping the interaction time short (average of 8.62 seconds) without any instruction on how to use the application.

During the tests, most of the 124 participants involved were surprised by the possibility of such application and were very receptive to its concept. Participants were also surprised by how simple it was to make it work. With that in mind, it is possible to assume that there is a lack for such application or another type of solution for such problem, i.e. helping tourists. We also observed this positive behavior in the first two studies where the majority of our participants were interested in installing the application in their own phone to try to use it when they were traveling.

One possible reason for such positive feedback, could be that when asked to comment about their experience with photos taken from strangers, all of them shared the same memories of quite a few frustrating photos. The major complain was that even if they did not liked the photograph they usually felt uncomfortable to ask for that stranger to shoot it again. The ones that actually did ask, also commented that sometimes even with a couple of different shots they could not get the final result that they were expecting.

Interestingly results from both the *tourist* participant and third party evaluators indicated that photos taken using our method were preferred over the ones without, however, our method focus is not on composition or image quality but rather on enforcing the desired vantage point created by the users. We posit that STC is complementary to other methods proposed in the HCI community that aim to help users better compose their travel photos. We posit that it could be paired particularly well with composition helpers ([16, 15]) since our method makes no assumption about the composition knowledge of the *tourist* user when s/he is setting the reference image. Computer assisted framing tools could then help the *tourist* to create a high quality reference frame (or a more visually appealing one) thus enhancing the final photograph while ensuring the desired vantage point.

In order to reduce interaction time we opted to use thresholds when matching the translation and rotation references of the mobile device as a 6DOF match would be particularly hard to achieve thus time consuming. Participants rated the reference and final photograph similar which indicates that a one-to-one match is indeed not needed. However, this can lead to small differences, or problems, in the final photograph. The difference is more prominent when the camera is not completely aligned with the horizon (as observed in the before and after depicted in Figure 6) than with the thresholds in the translation. Future refinement in the algorithm should prioritize the horizon’s line over the other two rotation axis.

After the user trials we observed that although most of the photos were taken in a large active public park, the scenes were not necessarily overly crowded, a common situation in some touristic sites. We informally tested the stability of

our technique in a busy open street market and observed no problems in re-creating the image although interaction time increased but was still under 20 seconds. Figure 11 shows two before and after photos from our informal tests.



**Fig. 11.** Reference (a,c) and final (b,d) photos for the tests in crowded spaces.

### 5.1 Limitations

The proposed technique has some limitations similar to other photography techniques that are based on computer vision techniques. Because it is in part video based it needs at least a partially lit scene in order to find the keypoints. We also do not address different focal lenses *i.e.* different zoom levels. Although zoom could be taken into account in the keypoint matching we assumed a fixed lens and locked the digital zoom feature in our application.

We opted to use global exposure measurement during our user trials because it is the default exposure method of point-and-shoot cameras. The problem with this method is that it can create dark, back-lit or unbalanced images when the light conditions are challenging. Although this problem is not related to our algorithm, exposure problems were present in two images taken by participant where one was part of the A-B testing by the external evaluators.



## 6 Conclusions and Future Work

In this paper, we introduced the concept of instant rephotography based on: existing techniques for rephotography, in the inertial sensors embedded in smartphones and automatic cameras, and in a specific need for suitable photos when one wants to take part of it and shall ask for the help of somebody else. With our technique it is possible to make use of rephotography with a lightweight and fast framework, and as a consequence, to deploy such techniques in mobile devices hence enabling a simpler and fast use.

Our method fits great to the tourist’s problem with their photographs. However, it is also suitable for many other applications, e.g. group and family photos. Our user experiments demonstrated that one reason for it to have worked so well in the travel photography scenario is the short interaction time and the UI intuitiveness and ease of use. Experiments also revealed that the assisted photographs are considered better than manual photos with statistical significance.



**Fig. 12.** Problematic images captured by STC: final image(a) of user #16 where it is possible to see exposure problems and final image(b) of user #9 that shows exposure problems with back-lightning.

Currently, the minimal amount of matched keypoints is set to one. Nonetheless, when a small number of matches is found, the system becomes confused leading to a greater time span before the automatic photography to be concluded. Further studies should access a minimum threshold, however we have not had problems during the user study regarding this matter.

As future work, we wish to create a method for the camera owner to set an area of interest so the camera can measure light correctly. The reason for that feature is to avoid problems with back-lightning where the subject is not correctly exposed. Such a problem can be seen in Figure 12.

Also, it is interesting to create an advanced shake avoidance system. We plan to add a post-processing stage that tries to minimize or remove image blur. As STC already has the IMU data at the time of the shot, it could save this information and use it in a deblurring stage, e.g. the one used by Joshi et al. [9].

## Acknowledgments

This study was partly funded by the Coordenacao de Aperfeicoamento de Pessoal de Nivel Superior - Brasil (CAPES) - Finance Code 001, and partly by CNPq. We also acknowledge FAPERGS (project 17/2551-0001192-9) and CNPq-Brazil (project 311353/2017-7) for their financial support. Thanks to all individuals that volunteered for the user tests.

## References

1. Adams, A., Talvala, E.V., Park, S.H., Jacobs, D.E., Ajdin, B., Gelfand, N., Dolson, J., Vaquero, D., Baek, J., Tico, M., Lensch, H.P.A., Matusik, W., Pulli, K., Horowitz, M., Levoy, M.: The frankencamera: An experimental platform for computational photography. In: ACM SIGGRAPH 2010 Papers. pp. 29:1–29:12. SIGGRAPH '10, ACM, New York, NY, USA (2010), <http://doi.acm.org/10.1145/1833349.1778766>
2. Bae, S., Agarwala, A., Durand, F.: Computational rephotography. *ACM Trans. Graph.* **29**(3), 24:1–24:15 (Jul 2010), <http://doi.acm.org/10.1145/1805964.1805968>
3. Balata, J., Mikovec, Z., Neoproud, L.: BlindCamera: Central and Golden-ratio Composition for Blind Photographers. *Proceedings of the Multimedia, Interaction, Design and Innovation* pp. 8:1—8:8 (2015). <https://doi.org/10.1145/2814464.2814472>, <http://doi.acm.org/10.1145/2814464.2814472>
4. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). *Comput. Vis. Image Underst.* **110**(3), 346–359 (Jun 2008), <http://dx.doi.org/10.1016/j.cviu.2007.09.014>
5. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: Brief: Binary robust independent elementary features. In: *Proceedings of the 11th European Conference on Computer Vision: Part IV*. pp. 778–792. ECCV'10, Springer-Verlag, Berlin, Heidelberg (2010), <http://dl.acm.org/citation.cfm?id=1888089.1888148>
6. Ercan, A., Erdem, A.: On sensor fusion for head tracking in augmented reality applications. In: *American Control Conference (ACC)*, 2011. pp. 1286–1291 (June 2011)
7. Hol, J.D., Schzn, T.B., Gustafsson, F., Slycke, P.J.: Sensor fusion for augmented reality. In: *FUSION*. pp. 1–6. IEEE (2006)
8. Jarusriboonchai, P., Olsson, T., Lyckvi, S.L., Väänänen, K.: Let's take photos together: exploring asymmetrical interaction abilities on mobile camera phones. *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services - MobileHCI '16* pp. 529–540 (2016). <https://doi.org/10.1145/2935334.2935385>, <http://dl.acm.org/citation.cfm?doid=2935334.2935385>
9. Joshi, N., Kang, S.B., Zitnick, C.L., Szeliski, R.: Image deblurring using inertial measurement sensors. In: *ACM SIGGRAPH 2010 Papers*. pp. 30:1–30:9. SIGGRAPH '10, ACM, New York, NY, USA (2010), <http://doi.acm.org/10.1145/1833349.1778767>
10. Lee, K.T., Luo, S.J., Chen, B.Y.: Rephotography Using Image Collections. *Computer Graphics Forum* (2011). <https://doi.org/10.1111/j.1467-8659.2011.02042.x>
11. Levoy, M.: Experimental platforms for computational photography. *IEEE Computer Graphics and Applications* **30**(5), 81–87 (2010)



12. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2. pp. 1150–. ICCV '99, IEEE Computer Society, Washington, DC, USA (1999), <http://dl.acm.org/citation.cfm?id=850924.851523>
13. McNulty, E.: Boston Then and Now. Thunder Bay Press (2002)
14. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(10), 1615–1630 (Oct 2005), <http://dx.doi.org/10.1109/TPAMI.2005.188>
15. Ni, B., Xu, M., Cheng, B., Wang, M., Yan, S., Tian, Q.: Learning to photograph: A compositional perspective. *IEEE Transactions on Multimedia* **15**(5), 1138–1151 (Aug 2013). <https://doi.org/10.1109/TMM.2013.2241042>
16. Rawat, Y.S., Kankanhalli, M.S.: Context-Aware Photography Learning for Smart Mobile Devices. *ACM Transactions on Multimedia Computing, Communications, and Applications* **12**(1s), 1–24 (2015). <https://doi.org/10.1145/2808199>
17. Rosten, E., Drummond, T.: Machine learning for high-speed corner detection. In: Proceedings of the 9th European Conference on Computer Vision - Volume Part I. pp. 430–443. ECCV'06, Springer-Verlag, Berlin, Heidelberg (2006), <http://dx.doi.org/10.1007/11744023.34>
18. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: Orb: An efficient alternative to sift or surf. In: Proceedings of the 2011 International Conference on Computer Vision. pp. 2564–2571. ICCV '11, IEEE Computer Society, Washington, DC, USA (2011), <http://dx.doi.org/10.1109/ICCV.2011.6126544>
19. Shingu, J., Rieffel, E., Kimber, D., Vaughan, J., Qvarfordt, P., Tuite, K.: Camera pose navigation using augmented reality. In: Mixed and Augmented Reality (ISMAR), 2010 9th IEEE International Symposium on. pp. 271–272. IEEE (2010)
20. Vazquez, M., Steinfeld, A.: An assisted photography method for street scenes. In: Proceedings of the 2011 IEEE Workshop on Applications of Computer Vision (WACV). pp. 89–94. WACV '11, IEEE Computer Society, Washington, DC, USA (2011). <https://doi.org/10.1109/WACV.2011.5711488>, <http://dx.doi.org/10.1109/WACV.2011.5711488>
21. Wang, Y., Song, M., Tao, D., Rui, Y., Bu, J., Tsoi, A.C., Zhuo, S., Tan, P.: Where2Stand. *ACM Transactions on Intelligent Systems and Technology* **7**(1), 1–22 (2015). <https://doi.org/10.1145/2770879>
22. Wen, J., Ünliür, A.: Redefining the fundamentals of photography with cooperative photography (Mum), 37–47 (2015). <https://doi.org/10.1145/2836041.2836045>
23. Woodman, O.J.: An introduction to inertial navigation. Tech. Rep. UCAM-CL-TR-696, University of Cambridge, Computer Laboratory (Aug 2007), <http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-696.pdf>
24. Yang, X., Cheng, K.T.T.: Accelerating surf detector on mobile devices. In: Proceedings of the 20th ACM International Conference on Multimedia. pp. 569–578. MM '12, ACM, New York, NY, USA (2012), <http://doi.acm.org/10.1145/2393347.2393427>