Laser Speckle Photography for Surface Tampering Detection

Edward Lu, <elu2@andrew.cmu.edu>
15-862 Fall 2022 Final Project

Abstract

For my final project, I implemented the paper "Laser Speckle Photography for Surface Tampering Detection," as well as ran additional experiments under different surfaces not shown in the paper. The paper aims to image the subtle changes on a surface given a pair of before and after images using a combination of computational photography and laser speckle imaging. Given nearly indistinguishable before and after images of a surface that has been touched or manipulated, the technique detailed in the paper is able to detect where the surface has been changed.

The paper leverages the fact that tiny surface deformations can cause small alterations in the speckle pattern of a laser projection. However, these alterations can only be found when the before and after images are perfectly aligned. The paper consists of a three step process. (1) Aligning the two images using a course alignment algorithm using image keypoints. (2) Aligning the two images using speckle-based alignment. (3) Computing the normalized cross-correlation of the two images to compute a similarity map.

Using a modified pipeline from the paper (skipping the course re-alignment step due to hardware restrictions), I am able to reproduce the results of its described algorithms and can successfully recover subtle surface changes given a pair of before and after images.

1. Introduction

In forensics, it is common to have the need to detect tampering on physical objects within a scene [7]. However, subtle forms of tampering such as hand/foot prints or small indentations caused by placing an object on top of another are extremely difficult to see with a naked eye, given a pair of before and after images. See Fig. 6 (c) and (d) and Fig. 1 (d)-(f) for an example. Detecting when a surface has been touched or tampered with is typically done with techniques using UV light or fluorescent powders. However, UV light only works well with bare-skin contact and non-porous surfaces, and fluorescent powders requires physically placing powder into a scene, which may leave the scene unable to be used for further forensics.

In order to detect the subtle changes on a surface when presented with a pair of before and after images, the authors of the paper *Laser Speckle Photography for Surface Tampering Detection* [14] attempt to leverage the speckle reflections generated by laser illumination. The paper exploits the fact that the speckle patterns from a laser are highly sensitive to small changes on an imaged surface. The idea is that when the travel distance of a laser changes (i.e., when it is beamed unto a surface and that surface's geometry changes), the phase of that laser changes. Since a laser speckle image encodes phase information (Fig. 2) [6], one should be able to recover exactly where the phase of a laser speckle image is different between two camera frames. For instance, the residues of the oils in your hand on a wall can cause a beamed laser's light wavefront to have subtle phase and intensity differences. If you take two laser speckle images before and after placing your hand on a wall, you should be able to detect where the phase is different between the two images, and thus be able to detect where your hand has touched the wall. See Fig. 6 (a), (b), and (f) and Fig. 1 (d) and (e) for reference.

The original paper leverages laser speckle imaging and out-of-plane modifications that come from small changes to a surface. The authors process two before and after images and apply normalized cross-correlation to recover the differences in laser speckle reflections, which can directly show a user where a surface has been tampered with (See Fig. 1 (c)). However, the laser speckles are very sensitive to the position of the camera, and directly computing normalized cross-correlation between two images of different viewpoints will not work; both images need to be aligned to the same viewpoint. The paper claims that these viewpoints need to be within half a millimeter for its algorithm to succeed. To align the before and after images, the paper first does a course alignment process using the PTAM [9] library, then does a fine alignment process using the speckles themselves. The authors use a motorized labjack to physically move a camera before taking an image so that the current image is aligned to the same viewpoint as the image taken before the surface has been tampered with.

For my project, I realized I did not need to apply the course alignment step described in the original paper in

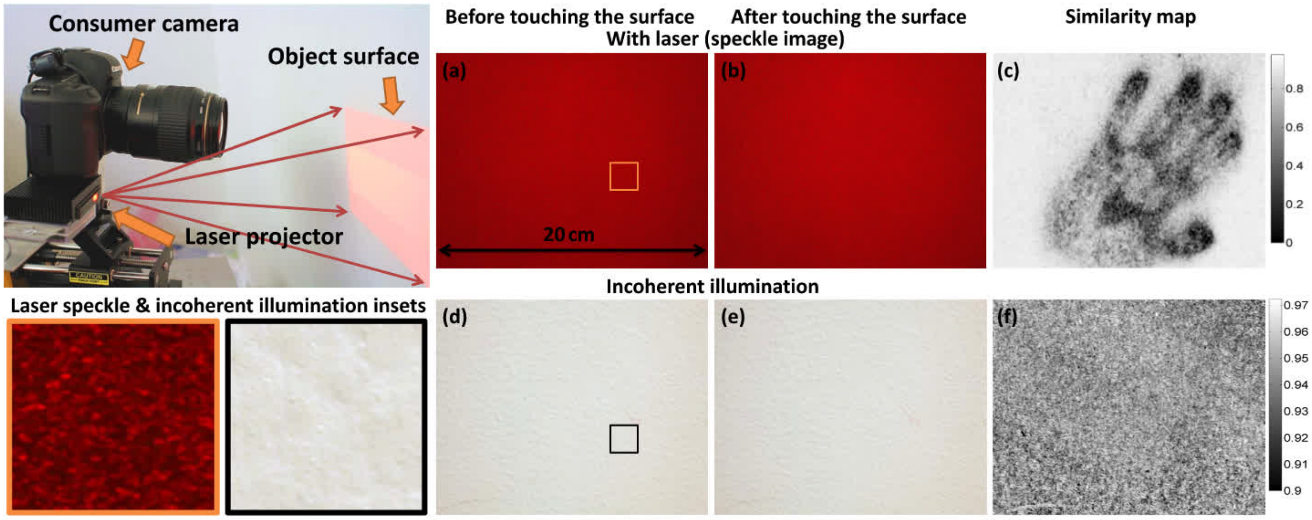


Figure 1: (Fig. 1 in the paper) The paper and my implementation both use an SLR with a pico laser projector. (a),(b) Images of a wall with a laser image projected on it. (a) is an image *before* the wall was gently touched by a hand. (b) is an image *after* the wall was touched. The speckle similarity map (c) reveals where the wall was touched. (d)–(f): Without the laser projector, the before and after images (d) and (e) reveal no difference, as shown in the similarity map (f). **Credit: Y. Shih, et al.**

order to explore its concepts. All that is needed are two images at the same viewpoint. So, I simply take two images before and after tampering with a surface at the same viewpoint of my camera. I implemented the algorithm described in the paper and have recreated its experiments as well as conducted my own experiments on various surfaces and types of tampering that have not been explored in the original paper.

The paper applies many topics covered in class, including radiometry and reflectance, coded photography, and lenses and optics.

1.1. Related Work

Laser Speckle Photography for Surface Tampering Detection My project is based on the algorithms and theories described in this paper [14]. The techniques work best for surfaces where tampering was subtle and where *only* the phase (and not intensity) of the laser reflection reaching the camera was changed.

Paper Authentication This technique uses the speckle reflections and roughness of a piece of paper to identify a unique signature [11, 10, 1, 3, 12, 7]. Like [14], the technique needs to ensure that the viewpoints of the before and after images of the piece of paper are sufficiently equal. The papers applying this technique typically have mechanical plates and mounts [12]. While [14] removes this requirement by using re-photography and speckle correction, my implementation still requires the same viewpoint of both the before and after images, which means a user cannot move the camera between frames.

In-plane motion sensing There are previous papers that project speckle patterns to present features on walls, transparent surfaces, or liquids to enable traditional object tracking and computer vision techniques [13, 6, 4, 2, 8].

Traditional Forensics These techniques may use fluorescent fingerprint powder, UV light, or other chemical methods. These usually only work for bare-skin contact (which means they can't detect non-skin objects placed on surfaces) and non-porous surfaces.

2. Background

I will now detail the theory behind why the before and after images need to have the same viewpoint.

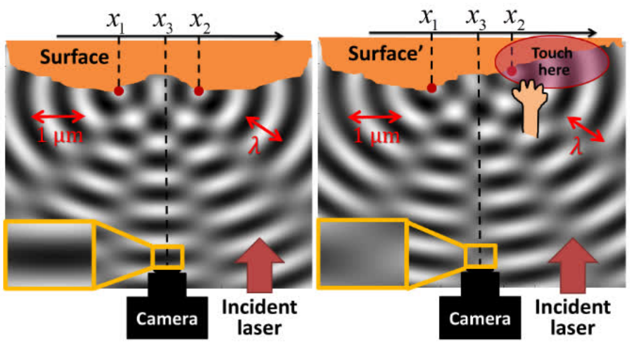
2.1. Speckle Image Formation

Speckle images can be modeled by the following equation [5, 6]:

$$I(y) = ||f(-\frac{z_1}{z_2}y) \otimes \tilde{g}(\frac{y}{z_2\lambda})||^2 \tag{1}$$

where $I(y)$ is the speckle image, and

$$f(x) = \frac{A_0}{z_1} e^{jk(h(x)+\frac{x^2}{2z_1})} \tag{2}$$



(a) (b)

Figure 2: (Fig. 2 in the paper) Touching a surface will change the waveform of the speckle reflections. (a) Shows the waveforms before touching a surface. (b) Shows the waveforms after touching the surface. Note that the surface heights, x_1, x_2, x_3, and the speckle image before and after touching the surface are different. **Credit: Y. Shih, et al.**

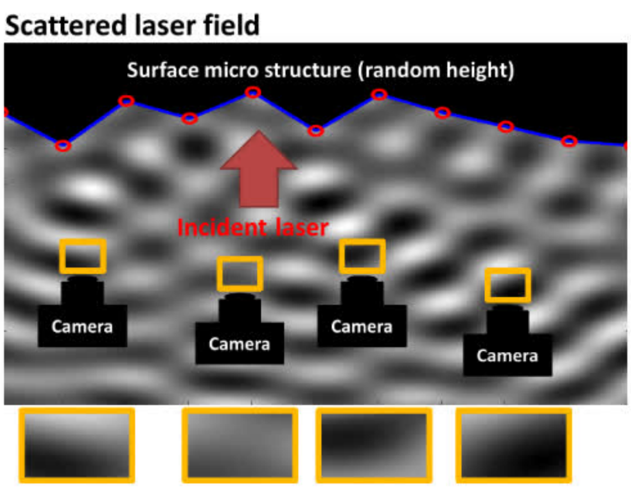
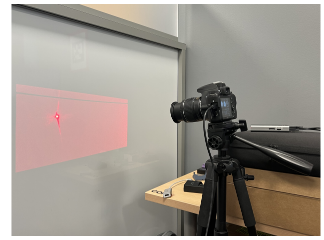


Figure 3: (Fig. 3 in the paper) The bottom row shows the images capture by the camera. Different viewpoints of the camera produce different speckle images. **Credit: Y. Shih, et al.**

where $h(x)$ is the surface height, x and y are the coordinates on the object and image plane respectively, A_0 is the incident wave amplitude, z_1 and z_2 are distances from the lens to the object and to the sensor respectively, and λ is the incident wavelength. $\otimes$ denotes convolution and $\tilde{g}(w)$ is the Fourier transform of the aperture function $P(u)$. **Credit: Y. Shih, et al.**

This equation shows that a change in $h(x)$ affects the image near $-\frac{z_1}{z_2}x$ because a finite aperture causes $\tilde{g}$ to fall

(a) Setup for wall experiment.



(b) Setup for whiteboard experiment.

Figure 4: Setup for experiments. I had a laser projector pointing at a surface, while a DSLR camera faced the projected laser image. I took two images before and after touching/manipulating the surface and an additional image for reference (for some experiments).

off rapidly. This means that the speckle image is affected by both camera translation and aperture size. So, a translation of the camera between taking a before and after image will drastically affect the speckle image. See Fig. 3 for a visualization of this effect.

3. Implementation

The original paper has three main steps:

1. Vision-based Computational Rephotography (either feature point mapping or structured light for flat surfaces).

2. Speckle-based Viewpoint Alignment.

3. Similarity Map Computation.

My implementation deviates a bit from the original paper and includes these three steps:

1. Conversion to Luminance Image.

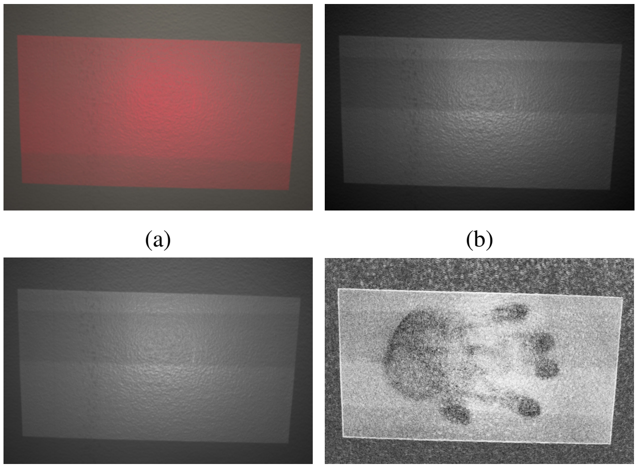
2. Speckle-based Viewpoint Matching.

3. Similarity Map Computation.

In this section, I will discuss how my implementation deviates from the original paper, as well as detail my full implementation.

3.1. Deviation from Original Paper

The authors of the original paper had a SLR camera and a micro laser projector mounted on a motorized labjack (See Fig. 1). The paper uses PTAM and the motors on the labjack to automatically and physically move the camera's view to



(a) (b) (c) (d)

Figure 5: Visualization of various steps in my implementation. (a) RGB image (after touching). (b) Luminance (Y-channel) image (after touching). (c) Luminance image after applying homography (after touching). (d) Resulting similarity map.

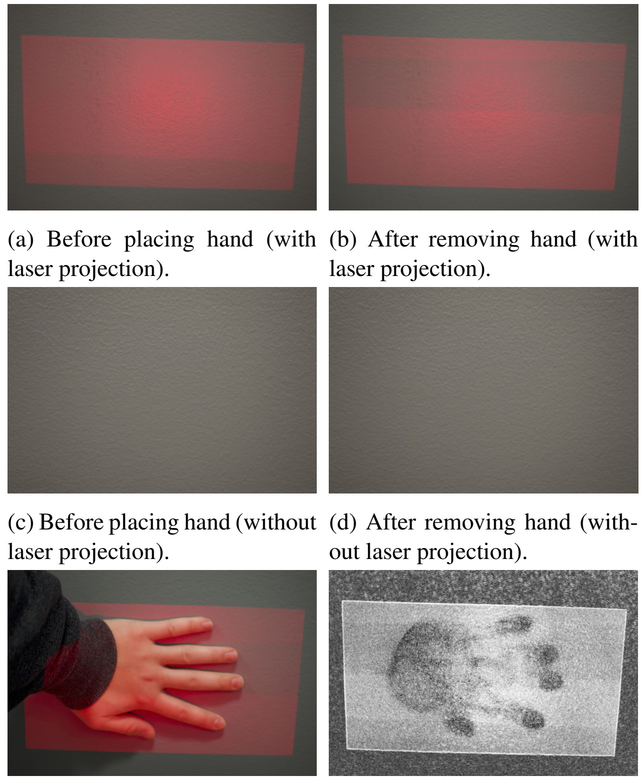
coarsely align with the original viewpoint of the surface in the before image. After this, it performs speckle-based realignment by translating the camera in-plane to ensure the same viewpoint between before and after images.

I did not have a motorized labjack, so I was unable to manually and accurately move the camera to the correct viewpoint before taking the after image. Thus, I did not implement the Vision-based Computational Rephotography or Speckle-based Viewpoint Alignment in the original paper, and instead simply ensured I took the after image at a similar viewpoint as the before image before performing speckle-based realignment. This means I could only tolerate about half a millimeter of movement between my before and after image. To do so, I kept my camera position static between shots.

3.2. Hardware

I used a Nikon 3500 (provided by the class) and a AnyBeam laser scanning pico projector (See Fig. 4). I projected a red image on the projector to match the figures in the original paper. I propped up the projector on a box and made sure there was minimal movement between the before and after images.

I took images in an indoor environment and used ISO values of 200-400 (depending on ambient lighting) with an aperture of f/5.6 (highest the camera could go to). The paper states that apertures between f/6.7 and f/16 worked the best experimentally for them, but f/5.6 (and even values below that) seemed to produce good images for me.



(a) Before placing hand (with laser projection). (b) After removing hand (with laser projection). (c) Before placing hand (without laser projection). (d) After removing hand (without laser projection). (e) (f)

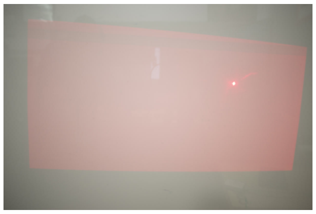
Figure 6: Experiment 1: hand on painted wall surface. (a),(b) Surface illuminated *with* laser projection before (a) and after (b) touching the wall. (c),(d) Surface *without* laser projection before (c) and after (d) touching the wall. (e) Hand touching the wall (for reference, not used in calculations). (f) Speckle similarity map (result of normalized cross-correlation after aligning both before and after images). The hand print is very clear, since the surface is not very reflective.

3.3. Conversion to Luminance Image

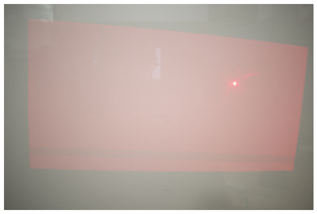
To simplify calculations to a single channel, I converted my before and after input images to the XYZ color space and grabbed the Y channel to extract luminance. I used the function provided to us in assignment 2 to convert to XYZ. See Fig. 5 for the conversion on an example image.

3.4. Speckle-based Viewpoint Matching

After taking the after image, I still needed to align both images in case there was some small rotation between shots. To do this, the original paper randomly sampled 100 patches in the before image I_{ref} and used normalized cross-correlation to find the corresponding patch in the after image I_{tar}. Then, they calculate a homography between I_{tar} and I_{ref} using the point correspondences and apply the homography to get a warped image $HI_{\text{ref}} = I_{\text{ref,w}}$. This makes

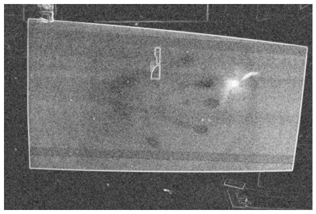


(a) Before placing hand.



(b) After removing hand.

(c)



(d)

Figure 7: Experiment 2: specular whiteboard. (a) Surface *before* touching the whiteboard. (b) Surface *after* touching the whiteboard. (c) Hand touching the whiteboard (for reference, not used in calculations). (d) Speckle similarity map (result of normalized cross-correlation after aligning both before and after images). Compared to Fig. 6, the hand print is less clear. This is due to the higher reflectiveness of the surface.

it such that the two images I_{tar} and $I_{\text{ref,w}}$ are as close as possible to being from the same viewpoint.

My implementation is the same as that of the original paper. For this, I used OpenCV's `matchTemplate` function, using the `TM_CCORR_NORMED` (normalized cross-correlation) method. Like the paper, I randomly sampled 100 correspondences of 50 pixel by 50 pixel patches between the two images and used OpenCV's `findHomography` (with RANSAC) to estimate the homography warp between the two images, then used `warpPerspective` to warp the reference image to match the target image. The paper does not specify the patch size they used for template matching, so I experimented with many patch sizes and found that 50 pixel by 50 pixel was the best size. See Fig. 5 for a slight warp on an example image.

3.5. Similarity Map Computation

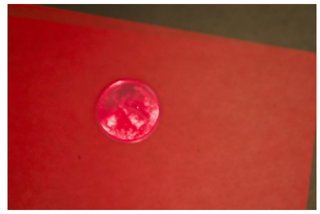
The similarity map is computed by calculating the normalized cross-correlation between the before image I_{tar} and the warped after image $I_{\text{ref,w}}$.

$$SCC(i,j) = NCC(W(i,j)I_{\text{tar}}, W(i,j)I_{\text{ref,w}}) \quad (3)$$

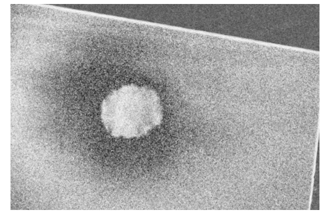
where $SCC(i,j)$ is the resulting similarity map at pixel (i,j) and NCC is the normalized cross-correlation function. $W(i,j)$ is a windowing function of size 21 by 21 at pixel (i,j).

(a) Before placing quarter.

(b) After removing quarter.



(c)



(d)

Figure 8: Experiment 3: placing a quarter on cardboard. (a) Surface *before* placing a quarter. (b) Surface *after* placing a quarter. (c) Quarter on the cardboard (for reference, not used in calculations). (d) Speckle similarity map (result of normalized cross-correlation after aligning both before and after images). The outline of the coin can be clearly seen.

To compute the similarity map, I took every 21 by 21 patch at every pixel and computed the normalized cross-correlation.

4. Results

I ran several experiments to test the robustness and to understand the limitations of my implementation. On my 2020 M1 MacBook Pro, my Python implementation takes around 15 minutes to recover the similarity map on the 4016 pixel by 6016 pixel RAW images from my DSLR camera. I have found that the higher resolution the image, the better the similarity map looks and the clearer the output. See Fig. 11 for a visualization of the pipeline of my system.

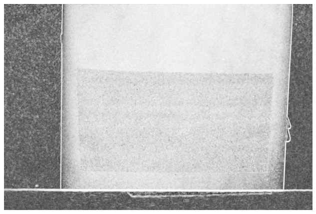
In my first experiment, I placed my hand on a wall to see if the similarity map is able to image where I placed my hand. Fig. 6 shows the before and after images and the results of this experiment. In the figure, I have also included before and after images without the laser projection to show that a human eye is unable to see the outline of my hand in the after image. This experiment shows that the system is able to very successfully image my hand on a painted gray wall.

The second experiment involved placing my hand on a more specular surface (a whiteboard). I wanted to see if the system can still image my hand when there are more reflections. Fig. 7 shows the results of this experiment. The hand is still visible on the similarity map, but less than that of the first experiment, which had a more diffuse surface.

(a) Before placing hand.

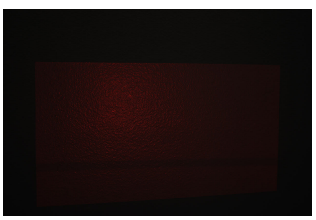
(b) After removing hand.

(c)

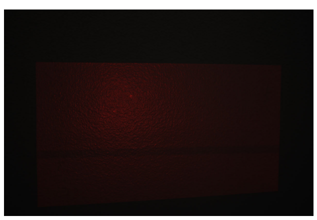


(d)

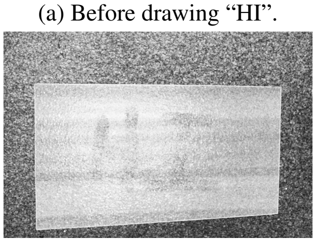
Figure 9: Experiment 4: placing hand on black felt fabric surface. (a) Surface *before* touching fabric. (b) Surface *after* touching fabric. (c) Hand on fabric (for reference, not used in calculations). (d) Speckle similarity map (result of normalized cross-correlation after aligning both before and after images). This experiment did not go as well as the others. This is due to the non-reflective nature of the felt material on the fabric not reflecting enough light from the laser projector.



(a) Before drawing "HI".



(b) After drawing "HI".



(c)

Figure 10: Experiment 5: drawing a "HI". (a) Surface *before* drawing. (b) Surface *after* drawing. (c) Speckle similarity map (result of normalized cross-correlation after aligning both before and after images). This experiment went well, you can see a faint outline of the letters "HI".

In the third experiment, I wanted to try the quarter experiment in the original paper (Fig. 7(a) in the original paper). In the paper, the authors placed a quarter (5.67 g) on a cardboard box and was able to recover the effect of the weight of the quarter on the box. Unlike the paper, I did not tape the coin to a string to avoid extra forces, so my similarity map does not reveal a very circular shape. Fig. 8 shows the results of this experiment. Overall, the experiment was successful. The figure shows a clear outline of a quarter and its effect on the cardboard box.

The original paper stated that the system fails for "volumetric" materials, such as carpets, sweaters, and scarves. So, I tried my system on a felt-like surface. Fig. 9 shows the results of this experiment. Like in the paper, this experiment was not very successful. The surface is not very well defined due to the material of the felt, and the material does not reflect much of the laser, so the similarity map does not show my hand.

I also wanted to try drawing with my finger, to see if my system can capture the strokes left by some of the oils from my finger. I took a before and after image of me drawing "HI" on a wall. Fig. 10 shows the results of the experiment. My implementation was able to successfully image the "HI."

The original paper claims that traditional forensics techniques only work for bare-skin contact, so I wanted to see if the laser speckle image-based technique detailed in the paper would work for non-skin related surface changes. Masking tape uses sticky material, and removing the tape will cause some of the material to stick to the surface the tape was on. So, I wanted to test if that sticky material can change the phase of the projected laser image, and if tape residue can be detected in the similarity map. In my sixth experiment, I taped a recognizable "X" shape using masking tape and took before and after images. The results of this experiment shows that the technique described in the paper *is* able to image tape residue. So, it can extend beyond just bare-skin contact on surfaces. See Fig. 12 for the results of the experiment.

The paper uses hands, fingers, feet, and scrapes on the wall in their experiments, but I wanted to see if it could also detect small changes in the wall that don't involve bare contact with skin or scraping of the surface. The paper shows that their system is able to detect fingerprints when a user is wearing gloves, but what if the user was covering their hand with their sleeve or with felt gloves? In my seventh experiment, I tried placing my hand on a wall, but covered with a sleeve. I pressed hard with my hand after placing the sleeve on the wall, hoping it would create subtle changes in the surface that would be detectable by a speckle comparison. Fig. 13 shows the results. The similarity map shows a tiny change in the area where my sleeve was, but it is not very noticeable. To a forensic analyst, it would seem like nothing has touched the surface. However, given the knowledge that a person touched the surface with a hand under a sleeve, one can make out a faint pattern where the sleeve was.

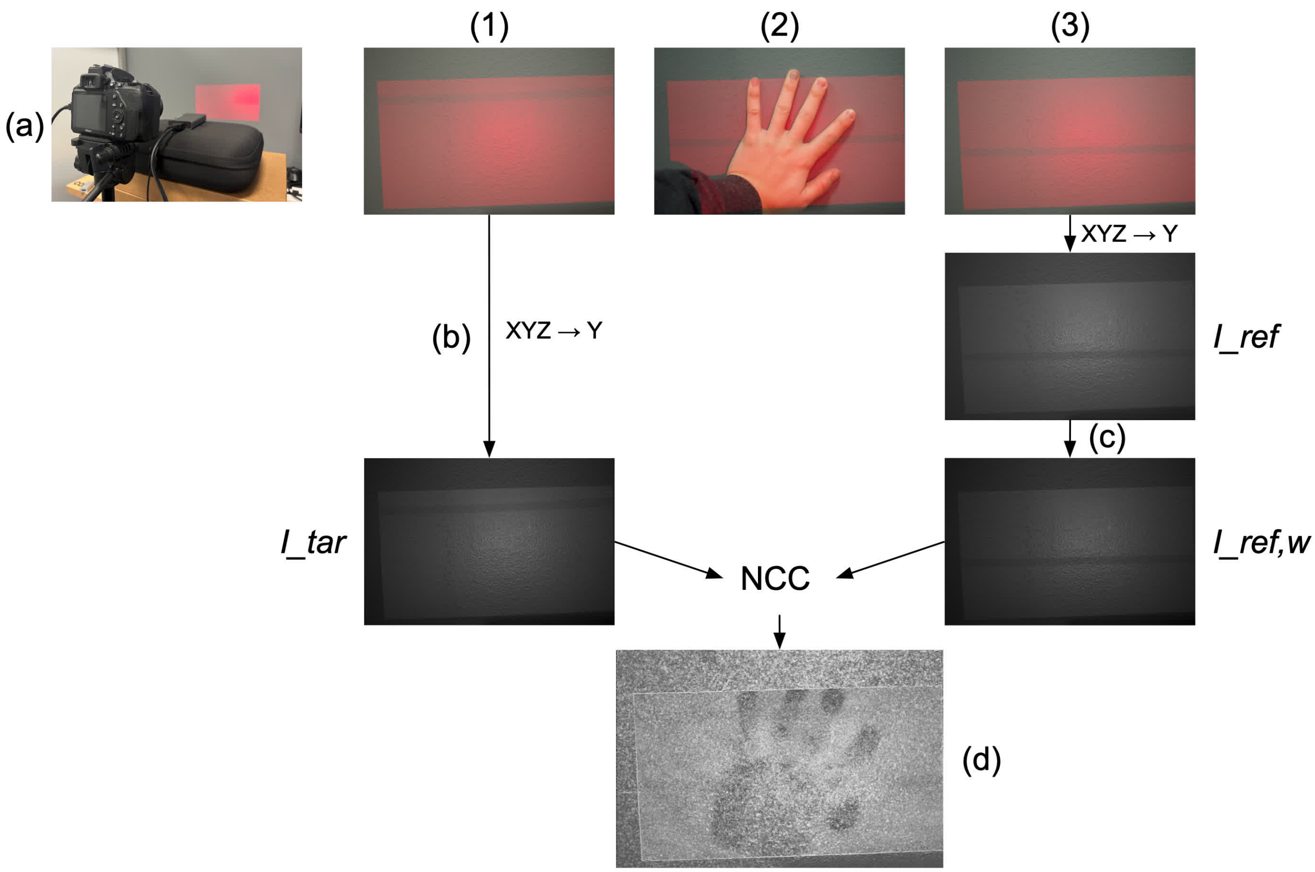


Figure 11: Pipeline of my experimental system. (a) I first take 2-3 images. (1) A before image, before I manipulate/touch the surface. (2) An intermediate image while I am manipulating/touching the surface, used for reference. (3) An after image, after manipulating/touching the surface. (b) After taking the images, I convert the before and after images into XYZ color space and grab the Y channel for luminance. (c) I randomly sample 100 corresponding patches between the before (I_{tar}) and after (I_{ref}) images to find a homography, apply the homography to align the viewpoints between the images, and find the warped $I_{\text{ref,w}}$. (d) Lastly, I use the normalized cross-correlation (NCC) to compute the similarity map between the two. If the experiment is successful, the resulting similarity map should present where the surface was manipulated/touched in a way that is viewable by the human eye.

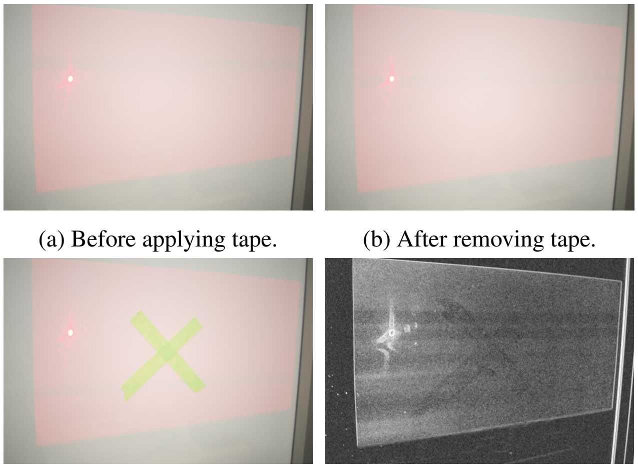
4.1. Limitations

My system suffers from similar limitations as that of the original paper. Fig. 7 shows that for reflective surfaces, the similarity map is not as clear. For highly reflective materials like mirrors or for highly transparent materials like glass, the technique will fail. Fig. 9 shows that for felt volumetric surfaces, the technique fails to image my hand. This is because the surface is not very well defined, as there are tiny strings of felt that can move before and after touching the surface, affecting the speckle of the laser. Additionally, the non-reflectiveness of the felt material did not reflect the laser, and therefore did not produce any speckle. The paper claims that "diffuse and moderately-glossy surfaces that are not too dark work best," and I have found that this is true.

Unlike the original paper, my implementation is fundamentally limited by the lack of automation in image viewpoint alignment. This affects the usability of my system, since the before and after image needs to align exactly, making it such that a user cannot move the camera between frames. Also, there is some aliasing in the speckle images captured by my camera (see Fig. 6 (a) and (b)) which can affect the normalized cross-correlation calculation. However, as long as the blank strips are not in the region of interest, it should be tolerable.

5. Conclusion

I have implemented and recreated the results of the paper *Laser Speckle Photography for Surface Tampering Detection* [14] for my final project. It details a technique



(a) Before applying tape. (b) After removing tape.

(c) (d)

Figure 12: Experiment 6: taping an "X" with masking tape on a whiteboard. (a) Surface *before* taping "X". (b) Surface *after* taping "X". (c) Tape on whiteboard in "X" shape (for reference, not used in calculations). (d) Speckle similarity map (result of normalized cross-correlation after aligning both before and after images). A faint "X" can be seen on the similarity map, showing the residue from the tape.

that can detect surface tampering for forensics or law enforcement applications using the speckle reflections of a laser. Using structured light and speckle correlation analysis, small surface changes can be detected given a pair of before and after images. I have also applied this technique to a number of new scenarios.

GitHub Repository with data and code here:
`https://github.com/EdwardLu2018/15862-final-project`
Video here:
`https://www.youtube.com/watch?v=e9CGhdTR1u0`

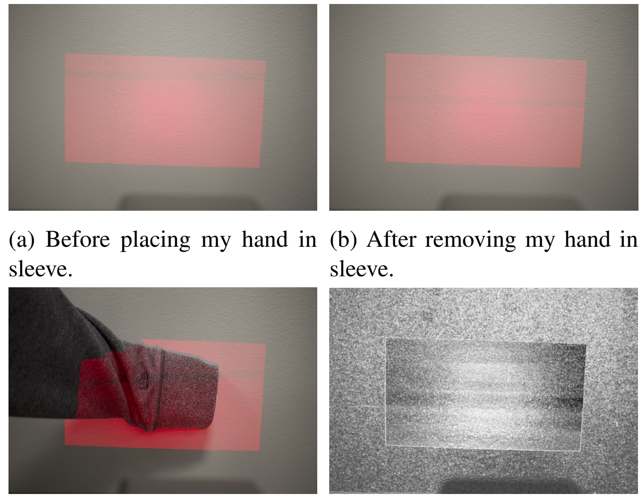
References

[1] L. Subramanian A. Sharma and E. Brewer. Paperspeckle. Paperspeckle: microscopic fingerprinting of paper, 2011. In *Proceedings of the 18th ACM conference on Computer and communications security*, pages 99–110. ACM, 2011. 2

[2] J. Briers and S. Webster. Laser speckle contrast analysis (lasca): a nonscanning, full-field technique for monitoring capillary blood flow, 1996. J*ournal of Biomedical Optics*, 1:174–179. 2

[3] R. Cowburn. Laser surface authentication-reading nature's own security code, 2008. *Contemporary Physics*, 49(5):331–342. 2

[4] A. Fercher and J. Briers. *Flow visualization by means of single-exposure speckle photography*, 1981. *Optics communications*, 37(5):326–330. 2



(a) Before placing my hand in sleeve. (b) After removing my hand in sleeve.

(c) (d)

Figure 13: Experiment 7: placing hand in sleeve on a wall. (a) Surface *before* placing hand in sleeve. (b) Surface *after* placing hand in sleeve. (c) Sleeve placed on wall (for reference, not used in calculations). (d) Speckle similarity map (result of normalized cross-correlation after aligning both before and after images). This experiment did not go as well, because the sleeve prevented the oils of my hand from going on the wall. Since there was no residue from the sleeve on the wall, the surface did not get deformed, and the system is unable to detect anything in the similarity map.

[5] J. Goodman. Statistical properties of laser speckle patterns, 1975. *Laser speckle and related phenomena*, pages 9–75. 2

[6] J. Goodman. *Speckle phenomena in optics: theory and applications*, 2007. Roberts Co. 1, 2

[7] A. Jausovec D. Petit P. Seem G. Xiong D. Atkinson K. Fenton D. Allwood J. Buchanan, R. Cowburn and M. Bryan. Fingerprinting documents and packaging, 2005. *Nature*, 436(28):475. 1, 2

[8] A. Olwal J. Zizka and R. Raskar. Specklesense: fast, precise, low-cost and compact motion sensing using laser speckle, 2011. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 489–498. 2

[9] G. Klein and D. Murray. Parallel tracking and mapping for small ar workspaces, 2007. In *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07)*, November 2007. 1

[10] J. Buchanan P. Seem and R. Cowburn. Impact of surface roughness on laser surface authentication signatures under linear and rotational displacements, 2009. Optics letters, 34(20):3175–3177. 2

[11] J. Taylor R. Pappu, B. Recht and N. Gershenfeld. Physical one-way functions, 2002. *Science*, 297(5589):2026. 2

[12] P. Ravikanth. *Physical one-way functions*, 2001. PhD thesis, Massachusetts Institute of Technology. 2

[13] R. Sirohi. *Speckle metrology*, 1993. CRC Press. 2

[14] S. Hasinoff F. Durand Y. Shih, A. Davis and W. Freeman. Laser speckle photography for surface tampering detection, 2012. In *Proc. 25th IEEE Conference on Computer Vision and Pattern Recognition*. CVPR 2012. 1, 2, 7