A Ground Truth Data Set for Two View Image Matching
IMM Technical Report 2010-05

Henrik Aanæs haa@imm.dtu.dk
Anders L. Dahl abd@imm.dtu.dk
Vess Perfanov vp@imm.dtu.dk

Department of Informatics
Building 321
Technical University of Denmark
2800 Kgs-Lyngby
Denmark

January 20, 2011

Contents

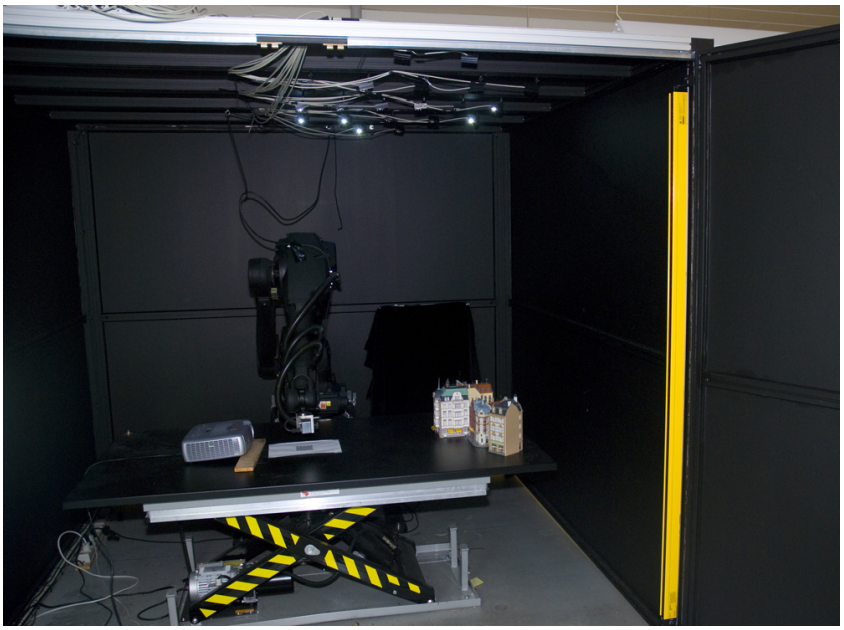


Figure 1: The DTU robot image capturing setup. It shows the camera mounted on the robot arm, the LED's in the ceiling used to control the scene lighting. The LCD-projector is used for structured light, and lastly the black box used for keeping all other light out.

1 Introduction

To facilitate a through evaluation of two view feature matching, we have compiled a large dataset of 135.660 images taken in a highly systematic and repeatable manner. The data set contains 60 complex and varying scenes, depicted from 119 positions and each position is illuminated from 19 individual LED light sources. In addition we obtained a 3D surface scan of the scene.

The camera is precisely positioned using an industrial robot. Scene illumination is varied systematically via a so called light stage – here 19 LED's mounted in black box together with the robot, see Figure 1. The 3D surface information for all 60 scenes was obtained via a structured light scanner. We used a stereo setup and structured light patterns to solve the correspondence problem. We have made the dataset publicly available from `http://roboimagedata.imm.dtu.dk` where the data can be freely downloaded. This report describes the data capture in detail. We are also in the process of doing in depth analysis of two view computer vision techniques, and the result of these studies will be published elsewhere. We will now provide an in depth description of the different parts of the setup, followed by a description of the depicted scenes and the procedure for image acquisition.

2 Robot Set Up

The purpose of our robot setup is to be able to control all aspect of the imaging process, to obtain systematic and repeatable measurements. The imaging process can be divided into the following three parts:

1. **Light** The light source(s) is the only reflected light from the scene.
2. **Camera** internal parameters, i.e. optics and position.
3. **Scene** The reflective properties and geometry of the object(s) being depicted.

The construction of these elements will be described in the following. A picture of our setup is shown in Figure 1, and a schematic drawing is given in Figure 2.

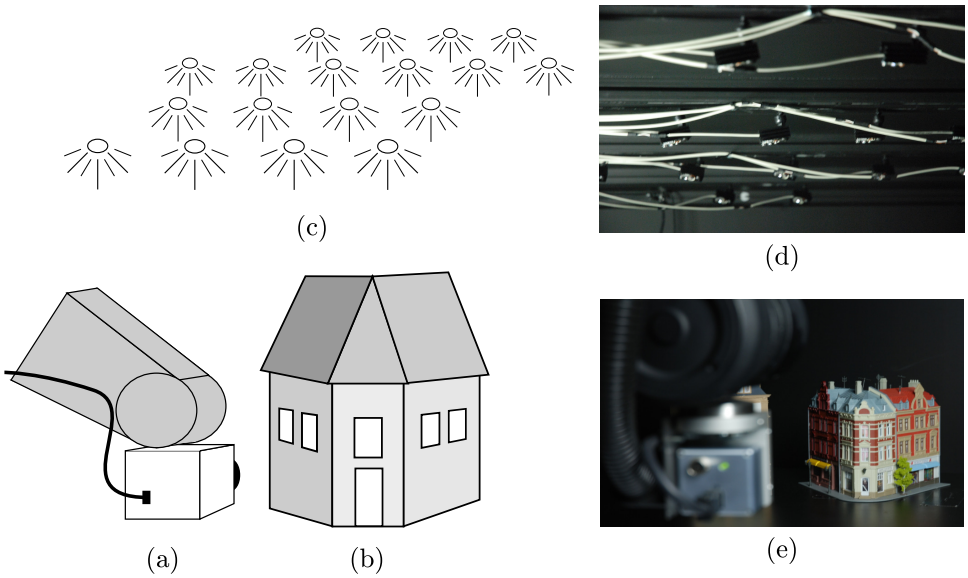


Figure 2: A schematic illustration of our set up. (a) is the camera mounted on the robot arm, (b) is the scene, (c) is the LED light stage, (d) is a photo of the LED light stage, and (e) is a photo of the robot arm and scene.

Figure 3: Images of our LED lighting setup. The spatial layout of the LED's is illustrated in Figure 4.

2.1 Light

To keep all external light out, our setup is encaged in a $2.5 \times 2.5 \times 2.1$m[1] closed box, which is painted with diffuse black paint on the inside, such that as little light as possible is reflected. Also the robot arm, which moves the camera, is painted black, such that the lighting conditions will be minimally effected by the robot position (unless it directly shadows a light, which typically can be avoided). Using this setup we ensure an illumination that only originates from the light stage. The result is a highly controlled illumination.

To have a flexible and controlled light setting, we built a light stage similar to the one described in Debevec *et al.* [2]. A light stage utilizes the physical property that two rays of light do not interact, and if we assume that the scenes only display linear reflective properties, light is additive. In practice this implies that two light sources, will result in the same image appearance if they illuminate the scene at once as if images are acquired individually and then averaged.

This is made into a practical device by setting up a large array of LED's (20 in our case[2]) as illustrated in Figure 3. An image is then taken with each of the LED's turned on individually, and then a virtual image can be composed through a weighted sum of the acquired images. E.g. a direct light setting can be achieved by just

[1] width×depth×height

[2] Only 19 were used in our experiment.

using the image from one LED, the direction can be varied by the choice of LED. A diffuse lighting can be achieved be making a relative uniform weighted average of all the images. See Figure 4.

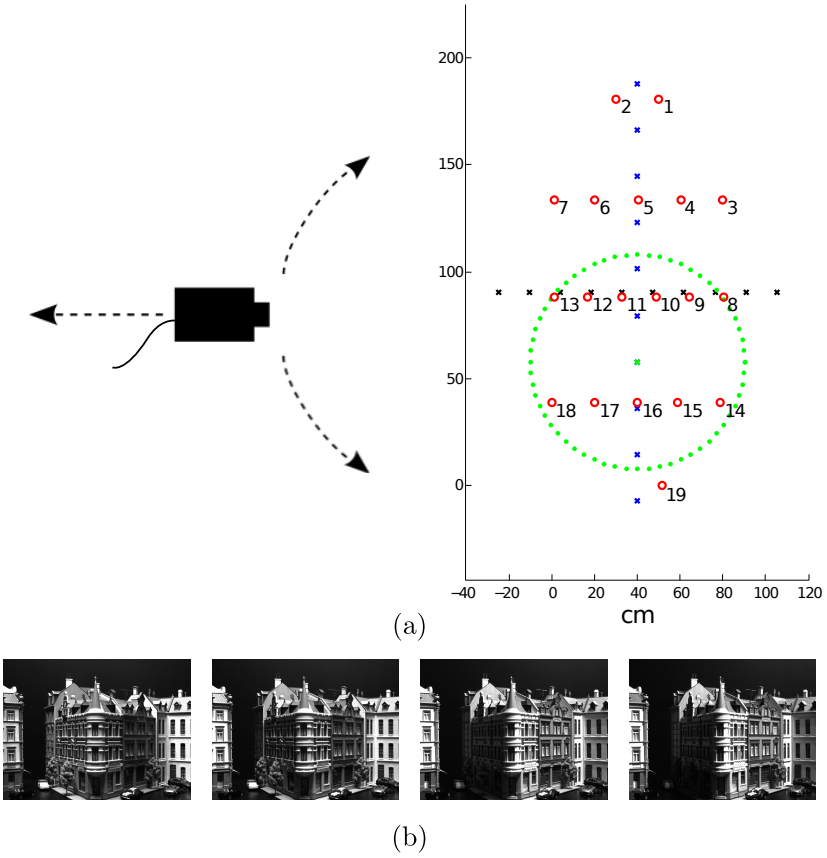


Figure 4: Layout of the light setup. (a) The camera (*left*) and LED's (*right*). The red circles are positions of LED's. Small blue and black markings show artificial light positions, and the green punctured marking illustrates a weighing area for the position at the green center dot. (b) Image examples of relighting from left to right.

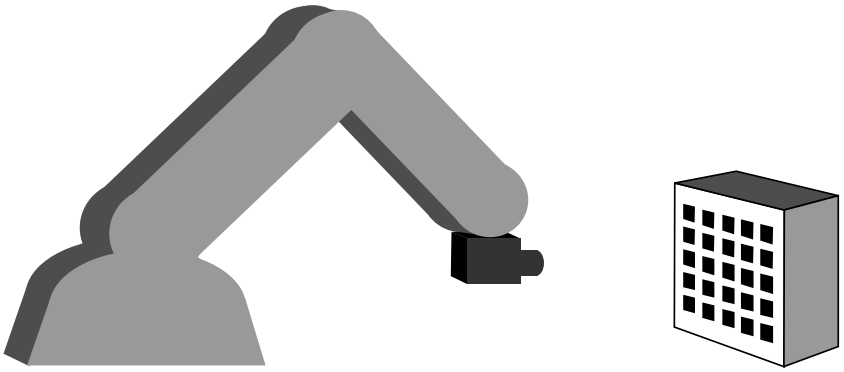
Relighting based on a weighted average of the images, will result in a reduction of white noise. An argument against this procedure is that the resulting images do not reflect the noise level of natural images. But in our setup we have aimed at high quality images and we have found the influence of white noise to be insignificant, even before averaging. Another question is if the linear relighting approach is applicable. If we have large nonlinear effects, for example from fluorescent materials, this could be an issue where the relighting might not be accurate. Given the use of the dataset, where we want a benchmark for systematically investigating algorithms for geometry and feature correspondence, we need images with a realistic appearance and variation. This is fully obtained from the proposed relighting scheme and the resulting relighting is highly adequate for this purpose.

2.2 Camera

The heart of our setup is an ABB IBR1600 robot where we have mounted a Point Gray Scorpion camera with a 1600×1200 color CCD chip, equiped with a 12mm lens. This is used for controlling the extrinsic camera parameters, i.e. position and orientation. Hereby the camera can be positioned reliably and repeatably.

The position accuracy of the robot is about 0.1mm, but the repeatability of the positioning is extremely small with an accuracy of about 0.05mm. We exploited the precise repeatability by calibrating a fixed set of camera positions, using the "Camera Calibration Toolbox for Matlab" by Bouguet[3]. The calibration setup is shown in Figure 5 Hereby we calibrated both the internal and external parameters.

[3] http://www.vision.caltech.edu/bouguetj/calib_doc/



(a) (b)

Figure 5: Schematic illustration of the calibration used in the robot. (a) The camera mounted on the robot was moved in front of a calibration object (b) with square pattern. Hereby both intrinsic and extrinsic parameters was estimated.

The repeatability accuracy of robot was estimated by comparing the positions obtained from the calibrations. The standard deviation of the camera positions is approximately 0.1 mm, and the standard deviation on the back projected pixel is 0.2 to 0.3 pixels. To obtain this accuracy we waited for four seconds at each camera position to ensure that no vibrations from the robot arm.

Lastly, it should be noted that we obtained raw sensor data from the camera. To account for additive noise originating from current leaks we had to normalize the images by subtracting an image acquired without light, but with the same settings as used for the dataset. We also corrected for radial distortion. It is these 'cleaned' images which we have used and made available.

2.3 Scene

The scene is controllable and we can place any collection of objects to form a scene in our setup, as long as it is not too big. An important property the ground truth 3D surface geometry of the scene. A potential image correspondence can be verified based on the known surface geometry, providing a strong tool in benchmarking computer vision algorithms.

We employed a structured light scanner to obtain the 3D scene surface [4]. Our setup is described in "Structured Light Scanner – Technical report" 2009, by Vesselin Perfanov[4]. It should be mentioned, that to get the 3D scene geometry in the same reference frame as the camera positions, we used four of the camera positions from our scene recording trajectory in the structured light scanner, c.f. Section 3.2.

We verify the precision of the structured light reconstruction using a white spherical object – a bowling ball painted with white diffusive light, and we measure the distance from the center of sphere to the surface. This gives an estimate of the surface reconstruction in the normal direction of the sphere. The advantage of a sphere is that it reveals error in all directions. We repeated the reconstruction of the sphere 10 times and we moved the projector between each scan. This gave a standard deviation of the radius estimate of 0.15 mm corresponding to a standard deviation of less than 0.6 pixels.

A precise method for obtaining a 3D surface is to use structured light, which we use in our experiments. Figure 6 (c) illustrates our surface reconstruction. We use a stereo camera setup and gray encodings to solve the correspondence problem. This method is recommended as one of the most reliable methods in both Scharstein and Pal [4] and Salvie *et al.* [3].

Some errors occur in the reconstruction and typically this is seen as individual points reconstructed from a wrong triangulation. To ensure a high quality of the 3D surface we have chosen to remove outliers. Removing points with too few neighbors does this. We removed points with less than 3 neighbors within a 0.5 mm sphere, retaining more than 99% of the original 3D points on average.

The scene surfaces have been scanned using two camera pairs at two distances from the scene. This is done to cover as much of the scene visible from the key frame as possible, see Figure **??**. The key frame is used as the reference image for evaluation, and in our experiments feature correspondence is found relative to that

[4] http://roboimagedata.imm.dtu.dk

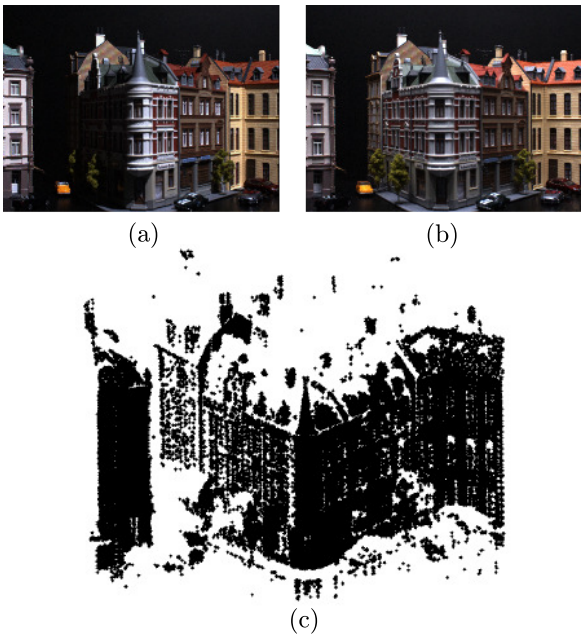


Figure 6: Image examples of the data set. An image with directional light from right (a), a diffuse image obtained by integrating all light directions (b), and the reconstructed 3D points (c).

frame. We obtain a varying number of surface points depending on the size of the scene from around 100.000 to 500.000 points.

3 Data Set Properties

As mentioned in the introduction we used the above described setup to photograph 60 scenes from 119 positions with 20 different light configuration (i.e. LED's). This dataset can be downloaded[5], This data is cite ware, so if you use please cite [1]. The bibtex reference is:

```
@conference{aanaes2010recall,
title={{On Recall Rate of Interest Point Detectors}},
author={Aan{\ae}s, H. and Dahl, A.L. and Pedersen, K.S.},
year={2010},
booktitle={3DPVT 2010:  Fifth International Symposium on 3D Data Processing,
Visualization and Transmission}}
```

In the following we will describe how the above described setup was used for this particular data set, specifically which scenes were chosen and from what positions they were photographed.

3.1 Scenes Chosen

A sample of the 60 different scenes uses are shown in Appendix A. In order to be able to do analysis on scene type as well most of the scenes are grouped into content classes as specified by Table 1. Hereby we could look for commonalities across scene types, and obtain results on the effect of scene type relative to the tested method. It should be noted that the 'Twigs and Leaves' scene class is noted to have a few very hard images in the dataset.

3.2 Position

119 images were taken for each scene in a horizontal plane approximately 10cm above the table where the scenes were placed shown in Figure 7. The results for the camera calibration can be found in the file `Calib_Results.ma`

[5] http://roboimagedata.imm.dtu.dk

Class	Scene Numbers	Total
House*	1, 4, 8, 31, 32, 49, 50, 55	8
Books*	2, 11, 20, 21	4
Fabric*	5, 6, 45, 46, 47, 48	6
Greens*	23, 24, 25, 26, 27, 51, 52, 53, 54, 56	10
Beer*	15, 16	2
Teddy Bears	9, 10 , 43, 44	4
Building Materials	33, 34, 35, 36, 37	5
Decorative Items (Art)	38, 39, 40, 41, 42	5
Groceries	12,28,29,30	4
Twigs and Leaves	17, 57, 58, 59, 60	5

Table 1: Most of the 60 scenes can be classified by topic, as seen here. The top five classes, denoted by a * and comprising half the data set, are the classes we made specific analysis on in our current analysis [1]. All the scenes have been used for the non-class specific analysis.

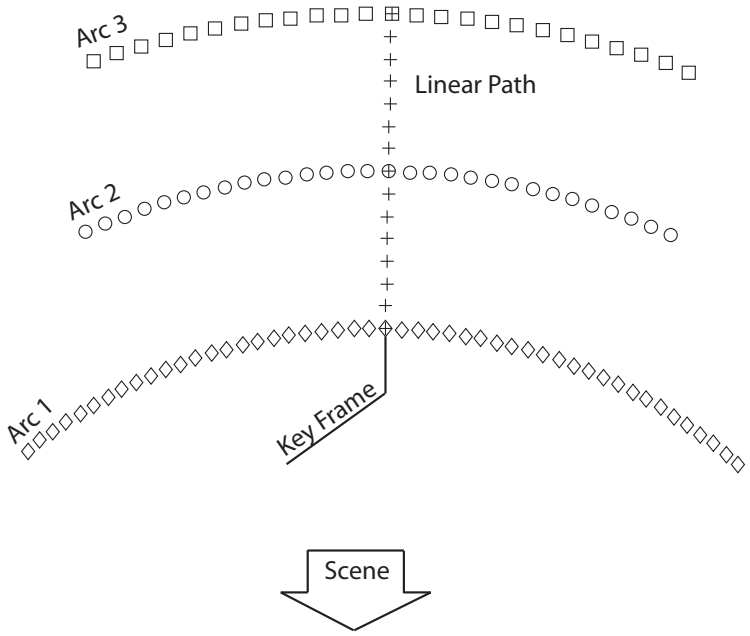


Figure 7: The positions of the camera relative to the scenes. The camera was always oriented such that it was pointing towards the scene, and such the rows of the image were horizontal, i.e. up in the image was up.

and file is generated with the "Camera Calibration Toolbox for Matlab"[6].

As seen in Figure 7, the camera positions consist of three arcs, centered in front of the scene, and a linear path moving away from the scene. Also it is noted that we have marked a frame as the *Key Frame* (frame number 25). The idea is that we evaluate two view correspondence algorithms between this frame and all the other (exactly a pair of view is needed for two view correspondence). This way we can compare the effect of angle at different scales, the distance to the scene via the two arcs, and the effect of just changing the distance or scale via the line.

4 Acquisition Protocol

For completeness we have included the scene acquisition protocol

1. Set up scene.

[6]Information about the calibration parameters and use of the toolbox can be found at: http://www.vision.caltech.edu/bouguetj/calib_doc/

2. Run “GotoCentalFrame” – check that it looks ok.
3. Make folder with name SET# with # equal to the data set number, e.g SET005. Make a subfolder named SL.
4. Make ”Notes.txt” with notes on scene.
5. Do structured light scan. See structured light note.
6. Ensure projector is turned off, as well as the flycap software and the LED terminal.
7. Run “GotoCentalFrame”.
8. Set camera parameters as follows:
   - Brightness 0
   - Exposure 2.41 ev
   - Gamma 1
   - Shutter 533.25 ms
   - Gain 13.21 dB
   - Frame rate 1.88

   WhiteBalance (should not have moved):
   - Red 90
   - Blue 80
   - Color processing method Rigorous (Slowest)
9. Validate that it looks OK.
10. Ensure the flycap software is turned off.
11. Set up ”LED_2View_Job.cpp” with
    - Image base name should be ”C://TwoView//Set#//Img”
    - DelayTime=4000
    - UseLight=true
12. Run “LED_2View_Job.cpp”
13. Goto 1.

5 Evaluation & Improvements

This is a pilot project for our robot setup. We are using this dataset and the analysis based on it to evaluate computer vision methods. The obtained results are satisfactory and we are confident that this is a well suited methodology for investigating and improving two view image correspondences.

However we plan to make a larger and improved dataset for two view matching. Things we would like to improve include:

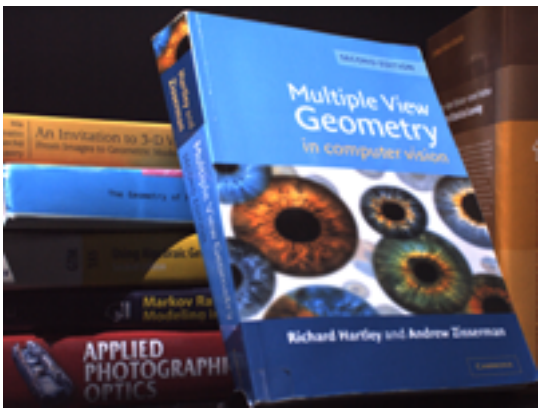
1. Have a larger number of scenes, preferably clustered in large scene type classes. Our investigations indicate that the variance of two view image correspondence methods across scenes is large, making it hard to draw statistically significant conclusions. A larger data set will reduce this variance and strengthen the conclusions.
2. We should also perform the structured light scan from more than one position, to obtain a better 3D coverage of the scene. At present the projector is located just behind the reference frame, which is adequate for the use in two view geometry. However, many of the occluding boundaries could be covered a bit better. This would also increase the usability of the dataset.

A Scenes

A sample image from each of the 60 different scenes, with the same lighting, is illustrated in this appendix. The numbering is the scene numbers

1.

2.



3.

4.

5.

6.

7.

8.

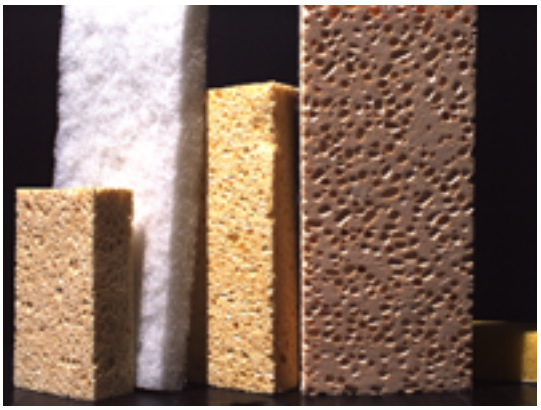
9.

10.

11.

12.

13.



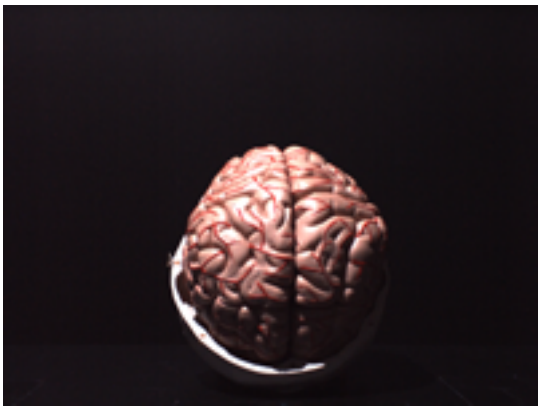
14.

15.

16.

17.

18.



19.

20.

21.

22.

23.

24.

25.

26.

27.

28.

29.

30.

31.

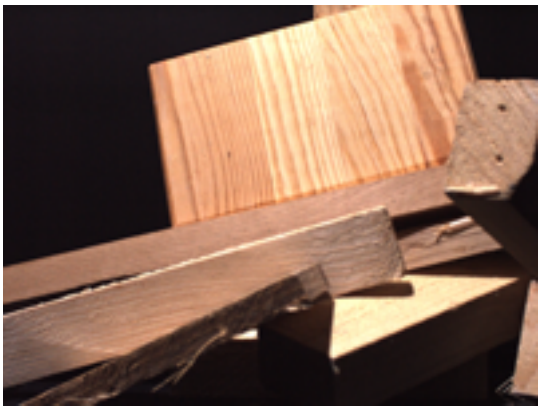
32.

33.

34.

35.

36.



37.

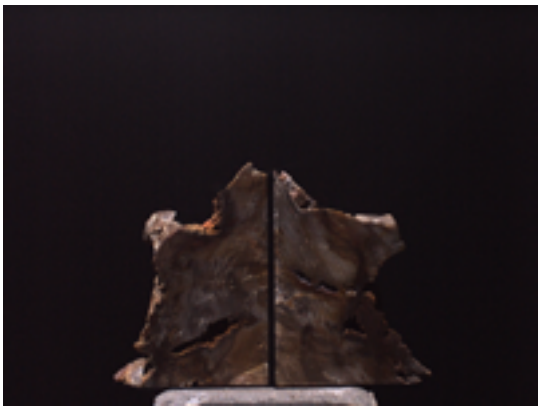
38.

39.

40.

41.

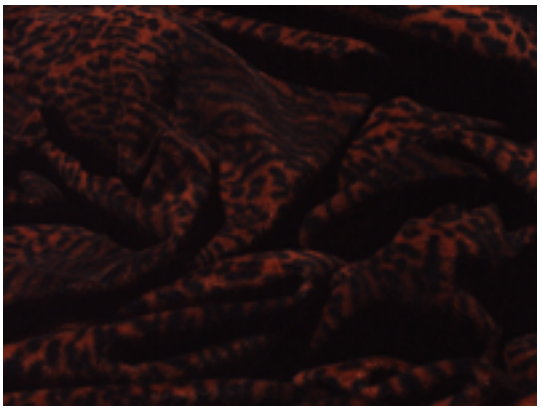
42.



43.

44.

45.



46.

47.

48.

49.

50.

51.

52.

53.

54.

55.

56.

57.

58.

59.

60.

References

[1] H. Aanæs, A.L. Dahl, and K.S. Pedersen. On Recall Rate of Interest Point Detectors. In *3DPVT 2010: Fifth International Symposium on 3D Data Processing, Visualization and Transmission*, 2010.

[2] P. Debevec, T. Hawkins, C. Tchou, H.P. Duiker, W. Sarokin, and M. Sagar. Acquiring the reflectance field of a human face. In *SIGGRAPH 2000*, pages 145–156, 2000.

[3] J. Salvi, J. Pages, and J. Batlle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37(4):827–849, 2004.

[4] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. In *CVPR*, volume 1, pages 195–202, 2003.