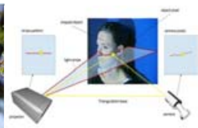
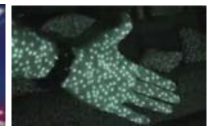




3D and Computational Photography

Prof. James Davis

Soc Sci 2 Rm 071 – MW 5:20-6:55pm

CMPS260 Computer Graphics is going to focus on 3D and Computational Photography in the Fall 2019 session. 3D cameras are integral to some of the hottest tech trends, for example AR/VR systems and self-driving cars. Your mobile phone takes pictures rivaling an expensive pro camera in large part because of the *computation* that happens whenever you take a picture.

You will be exposed to 80 research papers in this area, reading 40 and seeing another 40 in class.

You will do a research project and write a paper. In the past, some have published or been MS projects. We will explicitly use class time discussing how to perform and communicate research properly.

First 45 min of each class is – 'research topic lecture or paper presentation'

Second 45 min of each class is – 'logistics of class, how to write a paper, and the projects you are doing'

I expect the style of the class to appeal to PhD students and MS students interested in research.

Learning Objectives:

- Understanding of a sampling of classic 3D acquisition methods .e.g. Time of Flight, Stereo, Structured Light, Photogrammetry / Structure from Motion
- Understanding of a sampling of classic Computational Photography methods .e.g. Flash no flash, Panoramas, Lightfields, RTI, ML for images
- A broad but shallow survey of what is hot currently (all papers on these topics at SIGGRAPH in the last two years)
- How to perform and communicate research - picking topics, how to discuss related work, how to make a good figure, what is most important, how to review a paper, etc

Homework/grading:

- Read 4 papers a week (for exposure to research ideas) (20%)
  - Write short responses to 3 papers a week
  - Write a "review" to 1 paper a week
- Prepare presentations of research papers, about 10-15 minutes long (~2 times) (20%)
- Full quarter team research project – background research, code, experiment, write, (50%)
- Participate (come to class) (10%)

Topics:

- Basics: Camera: senors, bayer mosaic, assorted pixels, how cameras work, flash no-flash
- Basics: Image: RSME vs perceptual error metrics; color spaces YUV vs RGB

- Basics: Light ray: Lightfields, Coded aperture, Ptyography
- Project Background:Stereo3D: Stereo, Structured Light, ToF
- Project Background:ML: PCA, dim reduction, tex. synth, vector quant, CGAN, Image Analogies
- Project Background:RTI: PTM, Photometric Stereo
- Classic Application: Panoramas (10 years ago)
- Classic Application: Deblurring/denoising: Photo stacking vs Deconvolution (now)
- Classic Application: Background blur, refocusing (still works poorly)
- Paper writing: Mistakes not to make, sample review forms, what makes a strong paper
- *The above are classic topics, we will also have lots of recent papers as short presentations*

Homework details:

- Read 4 papers a week (for exposure to research ideas) (20%)
    - One picked by me – on the weekly topic area
        - Write a “review” following a standard conference review method
    - One more picked by me
    - Two picked by you – related work to the papers you are writing
        - Write a short ½ page journal entry including summary, what is the *contribution* of this paper, how did they *prove* it, what are your ideas for new papers
- Prepare presentations of research papers, about 10-15 minutes long (1-2 times during the qtr) (20%)
    - Most classes will have 2 papers presented by students
    - When its your turn, you’ll have a presentation to prep in addition to your journal
    - Your goal is to communicate the goal, the contribution, the results, and a high level of key methods in your 15 minute slot
    - Instructor rating of your presentation 10%
    - Your peers will be asked to rate your presentation 10%
- Full quarter team research project – background research, code, experiment, write, … (50%)
    - Background research – Search for existing papers to answer: Has someone done this exactly before? Has someone done something close? What are the categories of related work that already exist?
        - Turned in assignments, individual (10%)
    - Experiment/code – First make an explicit list of the results that you hope to obtain (before coding). Second, do what is needed to get these results.
        - Team portion of grade based on demos and results (10%)
        - Individual portion grade based on peer review from your team-mates (10%)
    - Write – We will talk about sections in a paper one at a time, using the class projects as examples. Expect homework of the form “produce a ‘Results’ section this week with fake draft figures showing what you hope to eventually get working for real”. Then we talk about those in class.
        - Individual writing 10%
        - Peer review of other peoples sections 5%.
    - Figures/captions – Figures, tables, plots, images all take much longer than words to get right. Expect multiple revisions as we talk about how to make these convincingly. They typically are the main “proof” that you have of your method working. 5%

- Participation (10%)
    - Come to class. Its expected, but missing when sick or you have a conference or interview is ok (you only lose 0.5%/day).

Class discussion and paper writing timeline:

Most classes first 1/2: Prof presents slides or leads discussion on 1 assigned paper, Students present 2 random modern papers
Most classes second 1/2: Prof presents or leads discussion on how to write this week's paper section.
Every class: Homework 1 assigned paper and 1 you pick (related to your project or presentation)
Homework related to paper writing to follow roughly the schedule below

Sep 30 –

- Intro research topics
- Intro syllabus, HW, paper writing and HW
- HW (2 day):: Find related work that might force a change to a project

Oct 2

- (research topics, exact order determined later according to project choices, repeats first ½ every class)
- Pick projects and teams
- HW (3 day): Find more related work, focus on the projects chosen

Oct 7

- Outline of all needed sections, tools, data, plots, to get a complete paper ready
- (change teams/projects if needed)
- HW (2 day): Make an outline plan for your paper, make an outline plan for internal deadlines you want to hit

Oct 9

- What makes a good figure, types of figures
- HW (3 day): Plan for results figures (completely fake, hand drawn, no real results)

Oct 14

- What goes in the related work, how to write this section defensively, bibliography
- HW (2 day):: Draft related work section (with whatever you know now)

Oct 16

- What goes in an introduction section, framing your contribution
- HW (3 day): Draft introduction and contribution statement (compatible with hypothesized related work and results)

Oct 21

- What goes in the method section, which things to describe and which to leave out
- HW (2 day):. Draft method section

Oct 23

- What makes a good figure part 2
- HW (3 day): At least one figure with real data (although maybe not the best result yet)

Oct 28

- What goes in the conclusion
- HW (2 day):: Conclusion section

Oct 30

- Putting it all together: Contribution -> Related work shows novel -> Results shows it works
- HW (3 day): Complete paper

Nov 4

- (complete paper is due)
- HW (2 day):: Review classmates papers

Nov 6

- (reviews of classmates due)
- HW (5 day): Fix comments from classmates
- HW: If submitting to CVPR, merge as needed

Nov 11 – Holiday

Nov 13

- (final paper is due)
- What goes in a video, common problems in videos, other supplemental material
- HW (5 day): Prepare a video, other supplemental

----Cvpr deadline Fri Nov 15 ----

Nov 18

-

Nov 20

- (video and supplemental due)
- In class: Comment on supplemental materials

- CVPR supplemental material deadline Nov 22

Nov 25 – Thanksgiving week (possible class canceled)

Nov 27 – Thanksgiving week (possible class canceled)

Dec 2

- What goes in a conference presentation
- HW (2 day): Prep your presentation

Dec 4

- Present your papers to the class

Possible Class Projects:

I've chosen three topic areas, and multiple sub-topics in each area. Why? Each sub-topic has a minor contribution that could be a workshop level paper or MS Project on its own. However the sub-topics are similar enough that they can be grouped together to write a stronger paper suitable for a high quality conference.

Why these topic areas? I've picked to satisfy the following criteria: possible to make a contribution in less than 10 weeks, theoretically easy enough that you don't need to read too many papers to get started, I have research experience in the area, there is real world impact available if we do a good job, there is a research publication possible, there is hands on system building available if you like that, there is theoretical work available if you like that, I can think of several related sub-topics to group together.

Are all these ideas really worth a paper? Heck no. These are ideas which *could* be worth a paper. I expect many of them have already been done. Part of research is figuring out the state of the art. We're going to do background research in the class. Some projects will have to be crossed out or modified. *The* most important part of research is getting the story straight, which often involves changing around how you present the contribution, even though its 90% the same thing underneath. These ideas are intentionally *raw*, so that we can go through the whole process together.

- Foundations: Stereo, spacetime stereo, structured light, photometric stereo
    - Stereo.1- Dual lens mobile with pico projector better than existing mobile 3D
    - Stereo.2- Pattern flash on dual lens mobile better than existing mobile 3D
    - Stereo.3- Dual structure light has advantage of no shadow
    - Stereo.4- What pattern to use - Stripe vs dot
    - Stereo.5 –Use the phase detect pixels for depth
- Foundations: PCA, ML for images, texture synthesis
    - ML.1 Machine learn all possible photos
    - ML.2 Machine learn noise characteristics high iso, (reduce chroma but not intensity)?
    - ML.3 Machine learn noise 3D cameras
- Foundations: RTI, self calibrate photometric stereo, histogram equalization and other norms
    - RTI.1– light position autocallibrate for mobile
    - RTI.2– light direction autocallibrate for mobile
    - RTI.3 – proj microscope
    - RTI.4 – proj - complete app and user test and processing speed, and internal capture
      RTI.5 – linear vs distinct points (artifacts)

Foundations: PCA, ML for images, texture synthesis

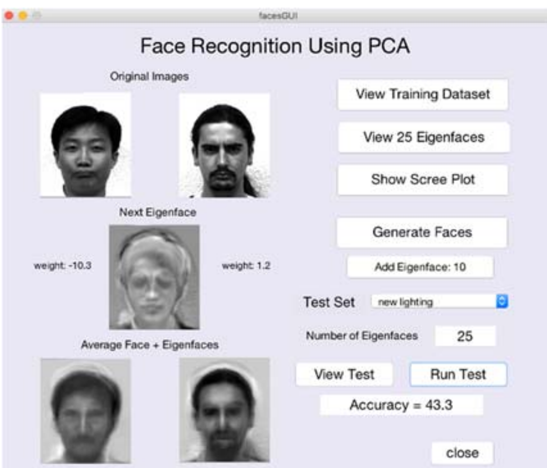
- **ML.1 Machine learn all possible photos**
- **ML.2 Machine learn noise characteristics high iso, (reduce chroma but not intensity)?**
- **ML.3 Machine learn noise 3D cameras**

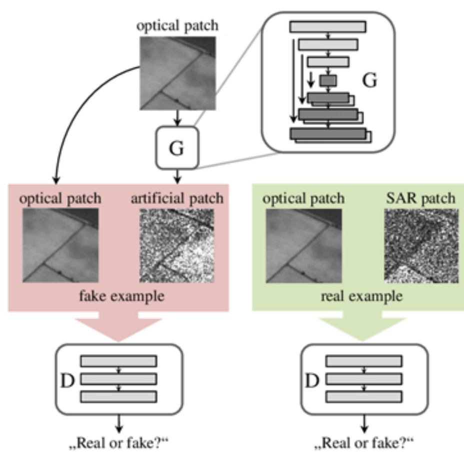
Why are these related?

The statistics of naturally occurring images lie in a low dimensional space. Its very unlikely to have a pure red pixel immediately adjacent to a pure blue pixel. Normally there should be a redish blueish pixel in between. We can use the information in a training set of images to tell us what to expect from future images. Any methods which attempt to learn this subspace and project the desired result from the available data might be applicable. This might be as simple as PCA, or vector quantization, or as fancy as this years ML favorite. Its reasonable that similar implementation would be applicable to all these projects.

Why is this topic in this class?

Machine learning on images is hot. All the cool kids are doing it. The specific goals chosen are all real problems with real application if we get a solution that works better than existing methods.





ML.1 Machine learn all possible photos

What is the goal?

- Suppose a camera is on a tripod and for the moment consider a static scene. You can twiddle ISO, shutter, aperture and a good photographer does. However the 1000 or so possible images possible are very related. Pixels just change brightness or focus level, but not their actual content. I think it should be possible to learn a model in which I take only 10 images and then just predict the rest. This would allow the photographer to twiddle these things *after* the image was taken rather than before. Or for an algorithm to auto-twiddle *after*.

What is the past work I know of now?

- Never seen the problem proposed like this. There are a bunch of existing methods that might be applied to actually learn the model. There are a bunch of ML papers showing much harder learning problems like predicting a photo from a sketch.

What is the contribution?

- Possibility 1) If we are the first, then showing a method that achieves this.
- Possibility 2) If someone has done it already, then we accept the lesser result of just being 10% more accurate.

How might this be implemented?

- To get datasets
  - Set up a scene and capture all possible photos, all 1000 or whatever. This is training/testing data.
- To run real time
  - A simple first version for static scenes would just take 10 photos rapid fire. A better solution would be to set up the sensor and lens to allow a single shot (an RGB bayar mosaic is a simple example of adjusting the sensor to get 3 channels of color in a single shot).
- To process the data
  - This might be as simple as PCA, or vector quantization, or as fancy as this years ML favorite.

Results?

- Show it works:
  - Images comparing ground truth to outcome
  - Table or curve with accuracy stats (RMSE)
  - Optional: User study to see if its possible to get photos preferred to the cameras default auto setting. Capture data with auto setting, let users pick the best settings post capture, then check which photos are rated higher.
- Analyze the method:
  - Quantifying the number of images needed vs accuracy.
  - Optional: Quantify along each axis focus, aperture, ISO, separately
- Limitations
  - Show and explain some failure cases

ML.2 Machine learn noise characteristics high iso, (reduce chroma but not intensity)?

What is the goal?

- Photographs taken in low light have very high noise. There should be less noise, as if the photo was taken in high light conditions. (This is the classic formulation)
- Alternate: Photographs taken in low light have very high noise. Noise is objectionable to viewers. The noise should be transformed to be less objectionable. For example in YUV space, denoising only in UV(chroma) and leaving Y(intensity).

What is the past work I know of now?

- There are 1000s of papers on denoising images. There is also work on de-blurring (needed if we let the shutter time be too long to get more light). Not sure if there are some using ML, or if ML will outperform other methods.
- Alternate: I've never actually heard it phrased this way in which its ok to leave noise, we just don't want it to bother the viewer.

What is the contribution?

- Someone has definitely worked on denoising, so our contribution is a method that is 10% more accurate.
- Alternate) If we are the first, then showing a method that has higher user preference.

How might this be implemented?

- To get datasets
  - Set up a scene and short and long exposure for training/test data.
  - Look on the web for existing datasets (since this is a known problem).
- To process the data
  - This might be as simple as PCA, or vector quantization, or as fancy as this years ML favorite.

Results?

- Show it works:
  - Images comparing ground truth to outcome
  - Table or curve with accuracy stats (RMSE)
  - Alternate: User study of image preference to show that its ok to just reduce one aspect of noise (like chroma). Need to establish that this is ok even though RSME remains high.
- Analyze the method:
  - ?
- Limitations
  - Show and explain some failure cases

ML.3 Machine learn noise 3D cameras

What is the goal?

- 3D cameras have a lot of noise in their measurements. This is both Gaussian noise around the correct depth and outlier noise with wildly incorrect depth on some pixels. It would be great to reduce either kind of noise.
- (I think the outlier noise is the most annoying, easy to show and possible to get rid of. The Gaussian noise is lower magnitude and harder to show. There is also overall bias in ToF cameras, and it would be very important to remove, but I'm not sure how to get a dataset with groundtruth).

What is the past work I know of now?

- There are 1000s of papers on denoising rgb images.
- I don't know of applying these RGB methods to 3D images.
- The outlier noise is normally removed through a variety of ad hoc filtering methods.

What is the contribution?

- If we are the first, then showing a method that works to reduce noise.
- If this is studied before, then showing that our method is 10% better.

How might this be implemented?

- To get datasets
  - o Find a 3D camera we can use, preferably ToF since its noise is largely random, as opposed to the consistent noise of a stereo setup.
  - o Set up a scene and short and long exposure for training/test data. Essentially averaging many shots will provide a 'noise free' version.
  - o Look on the web for existing datasets (?)
- To process the data
  - o This might be as simple as PCA, or vector quantization, or as fancy as this years ML favorite.

Results?

- Show it works:
  - o Images comparing ground truth to outcome
  - o 3D renderings comparing starting to outcome
  - o Table or curve with accuracy stats (RMSE)
- Analyze the method:
  - o ?
- Limitations
  - o Show and explain some failure cases
  - o

Foundations: Stereo, spacetime stereo, structured light, photometric stereo

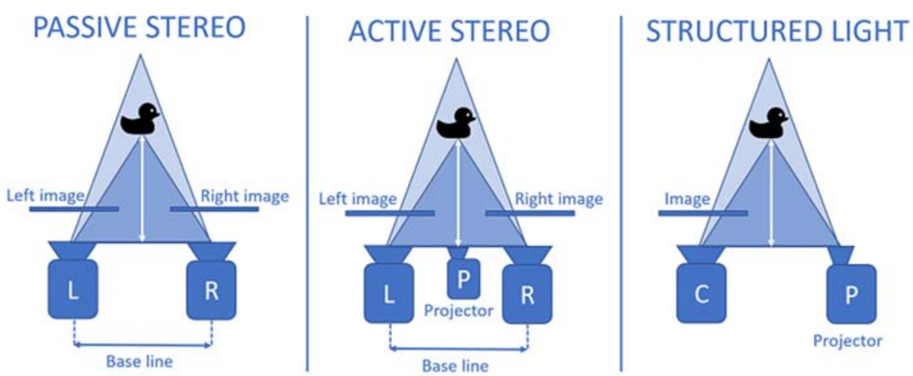
- Stereo.1- Dual lens mobile with pico projector better than existing mobile 3D
- Stereo.2- Pattern flash on dual lens mobile better than existing mobile 3D
- Stereo.3- Dual structure light has advantage of no shadow
- Stereo.4- What pattern to use - Stripe vs dot
- Stereo.5 –Use the phase detect pixels for depth

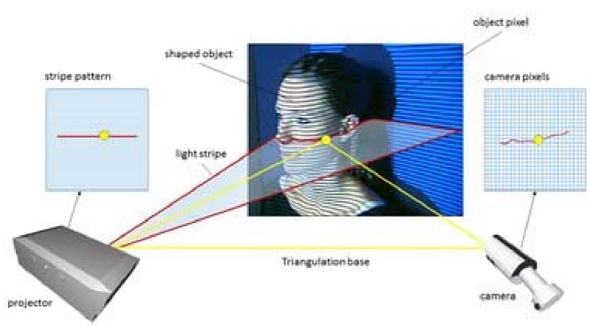
Why are these related?

All of these projects have a stereo 3D (triangulation) implementation at their core, but some new way to use the core algorithm.

Why is this topic in this class?

Triangulation (stereo) is the easiest 3D method to implement and the most flexible for software engineers. (ToF requires extensive hardware knowledge to modify). I believe we are going to have 3D on mobile devices, and it can fundamentally be much better than the current options. Mobile phones already contain most of the parts for good 3D, so the pathway to market is plausible. These projects all explore “improved” methods that might replace existing methods, and all are “practical” as opposed to theoretical. My own research and startup history matches this topic well.





Stereo.1- Dual lens mobile with pico projector better than existing mobile 3D

What is the goal?

- If mobile phones had 3D that would be useful. Existing mobile 3D cameras and apps are all of fairly low resolution (iPhone, ToF, etc). We want to show that a common dual lens camera on a mobile phone is capable of much higher accuracy 3D. There are structured light methods that use a sequence of patterns to capture 3D, and if the two cameras are calibrated, we don't need the projector calibrated. We use the pico projector to shine the patterns. All the parts are off the shelf, and we will show easy to capture 3D.

What is the past work I know of now?

- Mobile phones have 3D cameras – we will show higher resolution
- There are lots of stereo/structured light methods – we will use existing methods
- There are mobile apps doing structure from motion – these are of lower quality
- I don't know of anyone showing this working on mobile phone, only in laboratory conditions.

What is the contribution?

- We show that existing mobile phones are capable of much higher quality 3D with their existing hardware than the specialized 3D cameras being added to these devices.
- Practical: We need not make a whole working app to write a paper, but if we did, I think it would be a useful tool that would get used. For example, there is no practical way to scan an object for low cost 3D printing now. The printer is <$500, and good scanners are >$5000.

How might this be implemented?

- To get datasets
  - Need to be able to capture images from dual lens mobile phone camera. This is possible, but I do not know which phone, app, and/or library we should use.
- To process 3D
  - I suggest we first try a method called spacetime stereo. I have an implementation which is now ancient, but which we should be able to get running on a PC. (Written in 2003, but last ported and run in the 2015 time frame). We could write new code if needed.
- To get patterns
  - I hope to just load some patterns on a pico projector and turn it on.

Results?

- Show it works:
  - Images comparing depth from our method, and from iPhone 3D, and from existing 3D apps
  - Table or curve with accuracy on test target (want to claim at least 3x better than existing)
- Analyze the method:
  - Quantifying the number of lighting images needed vs accuracy.
- Limitations
  - Show and explain some failure cases

Stereo.2- Pattern flash on dual lens mobile better than existing mobile 3D

What is the goal?

- Mobile phones are starting to add 3D cameras, but they are adding special hardware to do that. These same phones already have dual lens cameras and a flash. Adding a patterned piece of plastic and a lens in front of the flash will allow 3D images to be taken with phones that do not have a specialized 3D camera.

What is the past work I know of now?

- Stereo is well studied
- Some phones have dedicated 3D
- Many phones have tried to get 3D from passive stereo (doesn’t work well).
- There have been some works showing how to make a “projector” from an LED, plastic, and lens

What is the contribution?

- Showing how to get better accuracy 3D from off the shelf phones, a simple clip on lens, and some software.

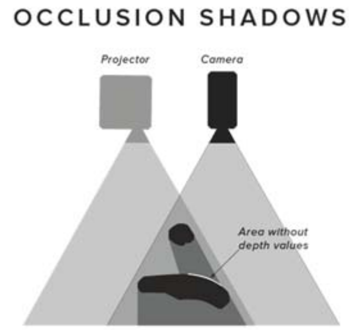
How might this be implemented?

- To get datasets
  - Modify a phone to make the flash pattern, and capture images with and without pattern.
  - Can we get the phone’s native depth map calculation?
- To process the data
  - Run any existing stereo method on the data with and without the flash patterns.

Results?

- Show it works:
  - Compare before and after 3D calculation from stereo
  - Table or curve with accuracy stats (RMSE)
- Analyze the method:
  - Quantifying the distance the flash is effective at, and how this falls off
  - Optional: Compare patterns
- Limitations
  - Show and explain some failure cases

Stereo.3- Dual structure light has advantage of no shadow



What is the goal?

- One of the primary disadvantages of triangulation (stereo and structured light) vs ToF is occlusion shadows in the depth map. We want to show that in a dual structured light (active stereo) setup we can remove these occlusions, addressing one of the main limitations of this method. We will essentially build a depth map from each camera/projector pair and merge these maps to fill in the occlusions.

What is the past work I know of now?

- Lots of work on stereo and structured light.
- Should be plenty of papers that describe this limitation.
- Not aware of anyone showing how to avoid it.

What is the contribution?

- Showing that active stereo need not have occlusion areas with missing depth data.
- Method to achieve this processing.

How might this be implemented?

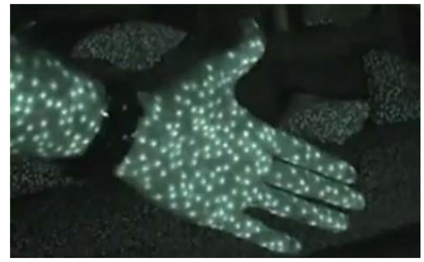
- To get datasets
  - Need to capture from a two camera + projector system like Intel Realsense. I think we can get access to one of these.
- To process the data
  - We need to modify a standard stereo setup to consider the projector like a 3rd camera. We have to invent some things to get the projector calibrated I think, but its close to standard stereo processing after that.

Results?

- Show it works:
  - Images comparing with and without shadow occlusions
  - Table or curve with pixel coverage stats
- Analyze the method:
  - ?
- Limitations
  - Show and explain some failure cases
  - Show the number of occluders or how complex a scene that there are still occlusions.
  -

Stereo.4- What pattern to use - Stripe vs dot

What is the goal?



- Active stereo (and structured light) project a pattern that will be used for stereo matching. All shipping systems use dot patterns while many academic papers use stripe patterns. We want to quantify the accuracy effects of changing pattern. (I predict stripes are actually better than dots, even though people use dots). Also look at robustness under slight mis-calibration.

What is the past work I know of now?

- Lots of papers using these things, not aware of anyone explicitly comparing patterns that are feasible to actually build into a cel phone.
- There are survey papers comparing various academic (mostly too hard in practice) methods.

What is the contribution?

- Analysis of the accuracy effects of changing pattern.

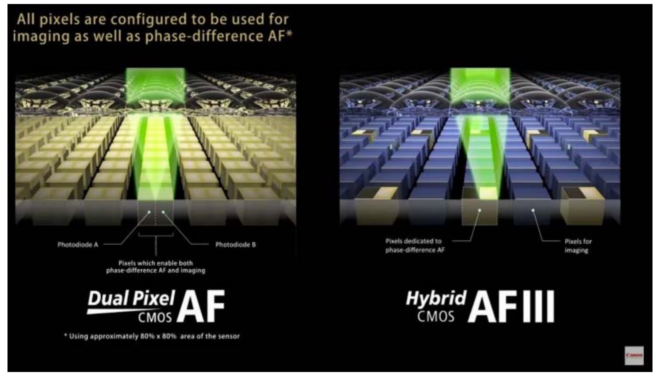
How might this be implemented?

- To get datasets
    - Need ground truth and various pattern datasets. Need a stereo camera and then can use any office projector to make patterns. We can use spacetime stereo to get the ground truth by shining lots of patterns. All the methods to compare are single pattern only.
- To process the data
    - Spacetime stereo or other stereo processing method. Should be able to avoid writing from scratch.

Results?

- Show it works:
    - Images comparing different patterns
    - Table or curve with accuracy stats (RMSE)
    - Compare accuracy when we intentionally miscalibrate
- Analyze the method:
    - Optional: Quantify changes in number of dots/stripes, randomness, and contrast
    - Accuracy as a function of amount of miscalibration
- Limitations
    - ?

Stereo.5 –Use the phase detect pixels for depth



What is the goal?

- Modern CMOS imagers have something called phase detect pixels used for auto focus. Some have every pixel this way. Essentially its splitting each pixel in two parts with separate microlenses and the two pixels get slightly different views of the scene. This is precisely the data needed to compute depth from stereo. The goal is to show that this is possible, and that its good for something. Since the baseline is small, the accuracy will be low, but that's probably ok.

What is the past work I know of now?

- There is lots of work on stereo
- I have seen an internal demo at Aptina showing that they have done this internally.
- I haven't seen a paper about this.

What is the contribution?

- Show that regular 2D cameras can already capture low accuracy depth because of the autofocus pixels they now contain.
- Optional: If this is used in a dual lens setup, show that the phasedetect pixel can be combined with the regular stereo to improve results.

How might this be implemented?

- To get datasets
  - Need a camera with these pixels that lets us save the raw measurements from both subpixels. I'm sure there is some camera that will let us do this if we look around.
- To process the data
  - Standard stereo algorithm.
  - Optional: could use a stripe or light projector to do laser scanning to improve accuracy

Results?

- Show it works:
  - Images showing depth
  - Table or curve with accuracy stats (RMSE) from some test object
- Analyze the method:
  - Quantifying the effect of aperture size vs accuracy.
  - Optional: Show only stereo, only phase detect stereo, and then combined
- Limitations
  - Show and explain some failure cases

Foundations: RTI, self calibrate photometric stereo, histogram equalization and other norms

- RTI.1– light position/distance autocallibrate for mobile
- RTI.2– light direction autocallibrate for mobile
- RTI.3 – proj microscope
- RTI.4 – proj - complete app and user test and processing speed, and internal capture
- RTI.5 – linear vs distinct points (artifacts)

Why are these related?

All of these projects relate to reflectance transformation imaging. In this method we take pictures without moving the camera, but changing the lighting location. The result is the ability to relight the image interactively as well as apply various analysis tools. Typically the resolution is higher than we can obtain with true 3D since its an image only technique.

Why is this topic in this class?

Museums and archeologists want to use this but devices are too expensive. I want to get this to run on mobile phones, so that it can be more widely used. There are several things that need to be solved to get this to work. Some are more research questions, and some more engineering. I also think the methods could be applied to microscopy. Working with museums and scientists in other fields is (a) very fun (b) leads to interdisciplinary impact you just can't have working on CS alone. Lastly the core methods are simple enough that I think we can get good results, even in an introductory class.




Light positioning setup for a horizontal RTI data capture. Image courtesy Carla Schroer

RTI.1– light position/distance autocallibrate for mobile

What is the goal?

- The RTI method requires that light sources be equidistant (or at least of known distance) from each pixel. In practice all deployed systems are domes with constructed fixed distance, or use a handheld string to keep the distance fixed. We wish we could make a mobile application in which you just wave around a light source for easy capture. But to do this we need to be able to normalize by light distance. Either by figuring out the light distance somehow, or by just normalizing the image. The goal being to construct RTIs without knowing the light distance a priori.

What is the past work I know of now?

- There is work on uncalibrated photometric stereo that solves for light direction/distance.
- There is a paper on light waving, which may have a solution.
- If we end up normalizing based on pixel values, then there are papers describing how to do that..

What is the contribution?

- Possibility 1) A new method of solving for light distance.
- Possibility 2) Showing a new method of RTI for which this is normalized. (If we use a known method to achieve this, we don't claim the method, we claim the application).

How might this be implemented?

- To get datasets
    - Need to capture data with both fixed and variable light distance so we can make comparisons.
- To process the data
    - Hopefully we can use the existing RTI tools, as opposed to write new code.
    - We will have to write something to normalize the images prior to inputing them to the existing solver.

Results?

- Show it works:
    - Images comparing ground truth (known lighting) to outcome (our solved lighting)
    - Table or curve with accuracy stats (RMSE)
    - Video showing why this is needed (the problems if we don't normalize)
    - Video showing that the problems go away
- Analyze the method:
    - Analyze how much variation we can handle and still get acceptable results.
    - Optional: Analyze how good a user can do at keeping distance fixed, to establish this is needed, but that the data lies in the range of error we can actually correct.
- Limitations
    - Show and explain some failure cases

RTI.2– light direction autocallibrate for mobile

What is the goal?

- The RTI method requires that light sources be in a known direction from each pixel. In practice all deployed systems are domes with constructed fixed direction, or use a mirrored sphere to get a measurement of direction. We wish we could make a mobile application in which you just wave around a light source for easy capture and don't have to use the sphere. But to do this we need to be able to find light direcion. The goal being to construct RTIs without knowing the light direction a priori.

What is the past work I know of now?

- There is work on uncalibrated photometric stereo that solves for light direction.
- There is a paper on light waving, which may have a solution.
- We may be able to do some sort of PCA/dimensional reduction to get a solution without actually having the real directions.

What is the contribution?

- Possibility 1) A new method of solving for light direction.
- Possibility 2) Showing a new method of RTI for which works. (If we use a known method to achieve this, we don't claim the method, we claim the application).

How might this be implemented?

- To get datasets
  - We can use existing datasets with known directions and just drop this information.
- To process the data
  - Hopefully we can use the existing RTI tools, as opposed to write new code for this part.
  - We will have to write something to find the light directions of course.

Results?

- Show it works:
  - Images comparing ground truth (known lighting) to outcome (our solved lighting)
  - Table or curve with accuracy stats (RMSE)
- Analyze the method:
  - Analyze how much lighting variation we can handle and still get acceptable results.
  - Optional: Analyze how good a user can do at covering all directions, to establish this is needed, but that the data lies in the range of error we can actually correct.
- Limitations
  - Show and explain some failure cases

RTI.3 – proj microscope

What is the goal?

- RTI has been used quite a bit for archeology items. The abilty to move the light and see small features seems to be to apply to microscopy as well. All we need for capture is a moving light around a microscope. The goal is to build something that works, and then talk to some scientists that use microscopes to see if its actually useful.

What is the past work I know of now?

- There has been some work on changing lighting for microscopes. They use a different terminology so not sure if its actually the same. We definitely need to learn about that.
- I am not aware of RTI/PTM formulation being used with microscopes.

What is the contribution?

- A new method of microscope imaging that allows seeing additional details.

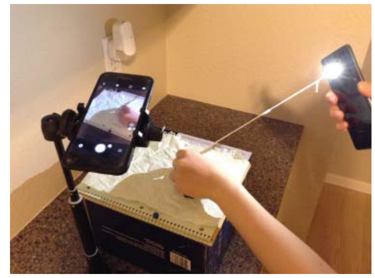
How might this be implemented?

- To get datasets
  - We'll have to obtain a microscope, either traditional of cel phone based. Then I think capture is just moving a light around and getting a set of images.
- To run real time
  - Not clear we have to run real time. I think we can do the proof of concept entirely offline and manually. We just need to prove its useful to apply this method in this domain.
- To process the data
  - Use existing tools for offline processing and viewing on PC.

Results?

- Show it works:
  - Images showing interesting things you can see
  - Table or curve showing some increase in ability to observe features. I'm a little unclear exactly what to measure. Have to brainstorm on this, but we need a table of numbers showing that this is better somehow.
  - Optional: Some sort of study together with real users.
- Analyze the method:
  - What kinds of materials does it work with?.
  - What is the minimum scale (max magnification) we can get this working with?
- Limitations
  - Show and explain some failure cases

RTI.4 – proj - complete app and user test and processing speed, and internal capture



What is the goal?

- RTI capture is currently slow. Either I use a $5K+ dome or I spend 20 minutes completing my capture. This should be a 30 second operation on a mobile phone. We wish we had an app to make this easy and quick.

What is the past work I know of now?

- I had a MS student do a project that developed a basic app. However it needs some improvements to be useful:
    - Clean up UI
    - Processing speed too slow. (The matrix operations are currently inefficient, we can definitely reorder to fix this)
    - It would be better if images were captured in the app, rather than imported
    - It would be great if there was export in a standard format
    - There are some things beyond the scope of this project, but we would integrate if they were solved by other teams:
        - Allow the user to freely hold the light and auto adjust for distance and direction.
- I am focused on the need in the archeology/museum space, however I've been told that there are some things happening in the gaming texture creation space that are related. We should at least learn about that.
- I am not aware of there being a mobile phone capture method.

What is the contribution?

- Mobile app and interface that makes it 10x faster to capture an RTI dataset.
- Optional: Showing that this is transformative for the users (who are currently skeptical because they worry about low mobile phone camera quality)

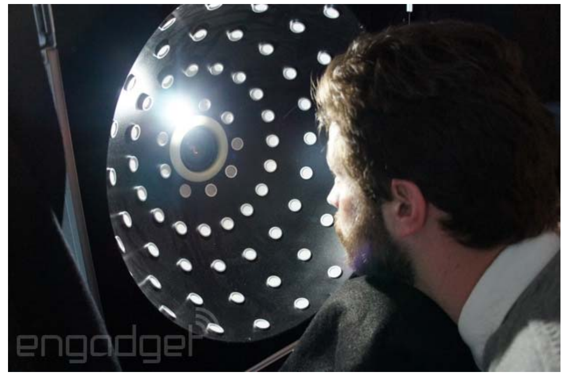
How might this be implemented?

- This is a coding project. Definitely need to modify the iOS code. I am not qualified to know if the last student left something clean, or a mess. You'd have to look.
- I'd be willing to have Android instead. The last student chose iOS, but I'm neutral.

Results?

- Show it works:
    - Images comparing traditional capture to mobile capture
    - Table or curve with accuracy stats (RMSE)
    - Table of numbers showing capture time speedup
    - Optional: User study to see what potential users think, and how it can be improved.
- Analyze the method:
    - Optional: Study on how camera quality affects final quality using synthetically degraded DSLR images to compare to ground truth.
- Limitations
    - Show and explain some failure cases

RTI.5 – linear vs distinct points (artifacts)



What is the goal?

- RTI works well (and is theoretically correct) for diffuse objects. However lighting with high frequency content like shadow edges and specular highlights show artifacts. In particular when you move the light around later you get a distinct "double image" feeling. This is undesirable and comes from the discrete lighting locations which are insufficient to sample the high frequency lighting effects (think signal processing and Nyquist limit). Anyway, the goal is to get rid of that artifact. We should be able to do that with larger area lights, but I think we can also try using "line sampling" instead of "point sampling". We move a light along a line and capture all 100s of images as we move it. Then process all this data together. I think the final result will then not have the artifacts.

What is the past work I know of now?

- I have never seen a paper that addresses this problem, and/or suggests anything except point samples for capturing lights.

What is the contribution?

- New capture method and showing that this method results in more pleasing results without artifacts.

How might this be implemented?

- To get datasets
    - Going to have to rig up a way to capture.
        - Manual – just painfully hold the light and move it to 100+ locations. Its going to take an hour or two, but its probably the easiest.
        - Automatic – I do have some controllable motors you could set up, which in my experience takes longer than just doing it, but does allow repeated experiments without pain, which is good.
- To process the data
    - Existing tools. Although with 100s of images, they might be a bit slow to run (which is ok for this project).

Results?

- Show it works:
    - Images comparing traditional to new sampling method, showing artifacts are gone
    - Optional: compare to using area lights
- Analyze the method:
    - How many lines are needed? Should we use radial lines, or concentric circles?
- Limitations
    - Show and explain some failure cases

Recent papers that someone in the class will present (as many as we can fit in the class)

Paper links from : http://kesen.realtimerendering.com/

Siggraph 2019

1. Image Science

Hyperparameter Optimization in Black-box Image Processing using Differentiable Proxies

Ethan Tseng, Felix Yu, Yuting Yang (Princeton University), Fahim Mannan, Karl St. Arnaud (Algolux), Derek Nowrouzezahrai (McGill University), Jean-Francois Lalonde (Universite Laval), Felix Heide (Princeton University and Algolux)

Handheld Multi-Frame Super-Resolution

B. Wronski, I. Garcia-Dorado, Manfred Ernst, Damien Kelly, Michael Krainin, Chia-Kai Liang, M. Levoy, P. Milanfar (Google Research)

A Unified Framework for Compression and Compressed Sensing of Light Fields and Light Field Videos

Ehsan Miandji, Saghi Hajisharif, Jonas Unger (Linkoping University)

Local Light Field Fusion: Practical View Synthesis with Prescriptive Sampling Guidelines

Ben Mildenhall*, Pratul Srinivasan* (UC Berkeley), Rodrigo Ortiz-Cayon (Fyusion Inc.), Nima Khademi Kalantari (Texas A&M), Ravi Ramamoorthi (University of California, San Diego), Ren Ng (UC Berkeley), Abhishek Kar (Fyusion Inc.) *denotes equal contribution

Synthetic Defocus and Look-Ahead Autofocus for Casual Videography

Xuaner (Cecilia) Zhang (UC Berkeley), Kevin Matzen (Facebook), Vivien Nguyen, Dillon Yao, You Zhang, Ren Ng (UC Berkeley)

9. Photo Science

Semantic Photo Manipulation With a Generative Image Prior

David Bau (Massachusetts Institute of Technology and MIT-IBM Watson AI Lab), Hendrik Strobelt (IBM Research and MIT-IBM Watson AI Lab), William Peebles, Jonas Wulff (Massachusetts Institute of Technology), Bolei Zhou (The Chinese University of Hong Kong), Jun-Yan Zhu, Antonio Torralba (Massachusetts Institute of Technology)

Progressive Color Transfer with Dense Semantic Correspondences (**TOG Paper**)

Mingming He (HKUST) Jing Liao (City University of Hong Kong), Dongdong Chen (University of Science and Technology of China (USTC)), Lu Yuan (Microsoft Research Asia), Pedro V. Sander (HKUST)

The Face of Art: Landmark Detection and Geometric Style in Portraits

Jordan Yaniv, Yael Newman, Ariel Shamir (The Interdisciplinary Center)

Distortion-Free Wide-Angle Portraits on Camera Phones

YiChang Shih, Wei-Sheng Lai, Chia-Kai Liang (Google)

14. Relighting and View Synthesis

Deep View Synthesis from Sparse Photometric Images

Zexiang Xu, Sai Bi (University of California, San Diego), Kalyan Sunkavalli (Adobe Research), Sunil Hadap, Hao Su, Ravi Ramamoorthi (University of California, San Diego)

Deep Reflectance Fields - High-Quality Facial Reflectance Field Inference From Color Gradient Illumination

Abhimitra Meka (Max Planck Institute for Informatics / Saarland Informatics Campus / Google), Christian Haene, Rohit Pandey (Google), Michael Zollhofer (Max Planck Institute for Informatics and Stanford University), Sean Fanello, Graham Fyffe, Adarsh Kowdle, Xueming Yu, Jay Busch, Jason Dourgarian, Peter Denny, Sofien Bouaziz, Peter Lincoln, Matt Whalen, Geoff Harvey, Jonathan Taylor, Shahram Izadi, Andrea Tagliasacchi, Paul Debevec (Google), Christian Theobalt (Max Planck Institute for Informatics and Saarland Informatics Campus), Julien Valentin, Christoph Rhemann (Google)

Multi-view Relighting using a Geometry-Aware Network

Julien Philip (Universite Cote d'Azur, Inria), Michael Gharbi (Adobe), Tinghui Zhou, Alexei (Alyosha) Efros (UC Berkeley), George Drettakis (Universite Cote d'Azur, Inria)

Single Image Portrait Relighting

Tiancheng Sun (University of California, San Diego), Jonathan T. Barron, Yun-Ta Tsai (Google Research), Zexiang Xu (University of California, San Diego), Xueming Yu, Graham Fyffe, Christoph Rhemann, Jay Busch, Paul Debevec (Google), Ravi Ramamoorthi (University of California, San Diego)

6. Scene and Object Reconstruction

Plan3D: Viewpoint and Trajectory Optimization for Aerial Multi-View Stereo Reconstruction **(TOG Paper)**

Benjamin Hepp (ETH Zurich), Matthias Niessner (Technical University of Munich), Otmar Hilliges (ETH Zurich)

Multi-Robot Collaborative Dense Scene Reconstruction

Siyan Dong (Shandong University), Kai Xu (National University of Defense Technology (NUDT)), Qiang Zhou (Shandong University), Andrea Tagliasacchi (University of Victoria), Shiqing Xin (Shandong University), Matthias Niebner (Technical University of Munich), Baoquan Chen (Shandong University and Peking University)

A Symmetric Objective Function for ICP

Szymon Rusinkiewicz (Princeton University)

Warp-and-Project Tomography for Rapidly Deforming Objects

Guangming Zang, Ramzi Idoughi, Ran Tao, Gilles Lubineau, Peter Wonka, Wolfgang Heidrich (KAUST)

Surface Reconstruction Based on Modified Gauss Formula **(TOG Paper)**

Wenjia Lu, Zuoqiang Shi Jian Sun, Bin Wang (Tsinghua University)

23. Video

Stylizing Video by Example

Ondrej Jamriska, Sarka Sochorova, Ondeej Texler (Czech Technical University in Prague), Michal Lukac, Jakub Fiser, Jingwan Lu, Eli Shechtman (Adobe Research), Daniel Sykora (Czech Technical University in Prague)

Video Extrapolation Using Neighboring Frames (**TOG Paper**)

Sangwoo Lee (KAIST and KAI Inc.), Jungjin Lee (KAI Inc.), Bumki Kim (KAIST), Kyehyun Kim (KAIST and KAI Inc.), Junyong Noh (KAIST)

Interactive and Automatic Navigation for 360Â° Video Playback

Kyoungkook Kang, Sunghyun Cho (DGIST)

Joint Direction and Stabilization for 360Â° Videos (**TOG Paper**)

Chengzhou Tang (Simon Fraser University), Oliver Wang (Adobe Systems Inc.), Feng Liu (Portland State University), Ping Tan (Simon Fraser University)

26. Computational Imaging

Coding Optimization for Fast Fluorescence Lifetime Imaging (**TOG Paper**)

Jongho Lee, Jenu Varghese Chacko, Bing Dai, Syed Azer Reza, Abdul Kader Sagar, Kevin W. Eliceiri, Andreas Velten, Mohit Gupta (University of Wisconsin-Madison)

Non-line-of-sight Imaging with Partial Occluders and Surface Normals (**TOG Paper**)

Felix Heide, Matthew O'Toole, Kai Zang, David Lindell, Steven Diamond, Gordon Wetzstein (Stanford University)

Wave-based Non-line-of-sight Imaging using Fast fk-Migration

David Lindell, Matthew O'Toole, Gordon Wetzstein (Stanford University)

Compact Snapshot Hyperspectral Imaging with Diffracted Rotation

Daniel S. Jeon, Seung-Hwan Baek, Shinyoung Yi (KAIST), Qiang Fu, Xiong Dun, Wolfgang Heidrich (KAUST), Min H. Kim (KAIST)

Siggraph 2018

(03) Computational Photography

Exposure: A White-Box Photo Post-Processing Framework (**TOG Paper**)

Yuanming Hu, Hao He (Microsoft Research and MIT CSAIL), Chenxi Xu, (Microsoft Research and Peking University), Baoyuan Wang, Stephen Lin (Microsoft Research)

Deep Examplar-Based Colorization

Mingming He* (HKUST), Dongdong Chen* (University of Science and Technology of China), Jing Liao (Microsoft Research Asia), Pedro V. Sander (HKUST), Lu Yuan (Microsoft Research Asia) (*Joint first authors.)

Locally Adaptive Rank-Constrained Optimal Tone Mapping

Xiao Shu Xiaolin Wu (McMaster University and Shanghai Jiao Tong University)

Deep Context-Aware Descreening and Rescreening of Halftone Images

Tae-hoon Kim (â€ŽIntel Corporation), Sang Il Park (Sejong University)

Non-Stationary Texture Synthesis by Adversarial Expansion

Yang Zhou, Zhen Zhu (Shenzhen University), Dani Lischinski (The Hebrew University of Jerusalem), Xiang Bai (Shenzhen University), Daniel Cohen-Or (Tel Aviv University) Hui Huang (Shenzhen University)

(08) Computational Photos and Videos

Synthetic Depth-of-Field with a Single-Camera Mobile Phone

Neal Wadhwa, Rahul Garg, David E. Jacobs, Bryan E. Feldman, Nori Kanazawa, Robert Carroll, Yair Movshovitz-Attias, Jonathan T. Barron, Yael Pritch, Marc Levoy (Google Research)

Stereo Magnification: Learning View Synthesis Using Multiplane Images

Tinghui Zhou (University of California at Berkeley), Richard Tucker, John Flynn, Graham Fyffe, Noah Snavely (Google Research)

Gigapixel Panorama Video Loops (**TOG Paper**)

Mingming He, Jing Liao, Pedro V. Sander (HKUST), Hugues Hoppe (Microsoft Research)

An Omnistereoscopic Video Pipeline for Capture and Display of Real-World VR

Christopher Schroers (Disney Research), Jean-Charles Bazin (KAIST), (Disney Research)

(20) 3D Capture

Space-time Tomography for Continuously Deforming Objects

Guangming Zang, Ramzi Idoughi, Ran Tao, Gilles Lubineau, Peter Wonka, Wolfgang Heidrich (KAUST)

Instant 3D Photography

Peter Hedman (University College London), Johannes Kopf (Facebook)

Reconstructing Scenes with Mirror and Glass Surfaces

Thomas Whelan (Facebook Reality Labs), Michael Goesele (Facebook Reality Labs and TU Darmstadt), Steven J. Lovegrove, Julian Straub, Simon Green (Facebook Reality Labs), Richard Szeliski (Facebook), Steven Butterfield, Shobhit Verma, Richard Newcombe (Facebook Reality Labs)

Full 3D Reconstruction of Transparent Objects

Bojian Wu, Yang Zhou, Yiming Qian (Shenzhen University), Minglun Gong (Memorial University of Newfoundland), Hui Huang (Shenzhen University)

Object-Aware Guidance for Autonomous Scene Reconstruction

Ligang Liu, Xi Xia, Han Sun, Qi Shen (University of Science and Technology of China), Juzhan Xu, Bin Chen, Hui Huang (Shenzhen University), Kai Xu (National University of Defense Technology (NUDT))

(23) Computational Cameras

What Are Optimal Coding Functions for Time-of-Flight Imaging?

Mohit Gupta, Andreas Velten (University of Wisconsin-Madison), Shree K. Nayar (Columbia University), Eric Breitbach (University of Wisconsin-Madison)

Single-Photon 3D Imaging with Deep Sensor Fusion

David B. Lindell, Matthew O'Toole, Gordon Wetzstein (Stanford University)

End-to-end Optimization of Optics and Image Processing for Achromatic Extended Depth of Field and Super-resolution Imaging

Vincent Sitzmann Steven Diamond, Yifan Peng, Xiong Dun, Stephen Boyd (Stanford University), Wolfgang Heidrich (KAUST), Felix Heide, Gordon Wetzstein (Stanford University)

Megapixel Adaptive Optics: Towards Correcting Large-scale Distortions in Computational Cameras

Congli Wang, Qiang Fu, Xiong Dun, Wolfgang Heidrich (KAUST)

Siggraph Asia 2018

Acquiring and Editing Geometry via RGB (D) Images

CurveFusion: Reconstructing Thin Structures from RGBD Sequences

Lingjie Liu*, Nenglun Chen* (The University of Hong Kong), Duygu Ceylan (Adobe Research), Christian Theobalt (MPI Informatics), Wenping Wang (The University of Hong Kong),Niloy Mitra (University College London)

Semantic Object Reconstruction via Casual Handheld Scanning

Ruizhen Hu, Cheng Wen (Shenzhen University), Oliver van Kaick (Carleton University), Luanmin Chen, Di Lin (Shenzhen University), Daniel Cohen-Or (Shenzhen University and Tel Aviv University), Hui Huang (Shenzhen University)

The Need 4 Speed in Real-Time Dense Visual Tracking

Adarsh Kowdle, Christoph Rhemann, Sean Fanello, Andrea Tagliasacchi, Jonathan Taylor, Philip Davidson, Mingsong Dou, Kaiwen Guo, Cem Keskin, Sameh Khamis, David Kim, Danhang Tang, Vladimir Tankovich, Julien Valentin, Shahram Izadi (Google Inc.)

PAPARAZZI: Surface Editing by way of Multi-View Image Processing

Hsueh-Ti Liu, Michael Tao, Alec Jacobson (University of Toronto)

Real-time High-accuracy 3D Reconstruction with Consumer RGB-D Cameras

Yan-Pei Cao (Tsinghua University, Beijing), Leif Kobbelt (RWTH Aachen University) Shi-Min Hu (Tsinghua University, Beijing)

Image Processing

Deep Unsupervised Pixelization

Chu Han*, Qiang Wen*, Shengfeng He, Qianshu Zhu, Yinjie Tan, Guoqiang Han (South China University of Technology), Tien-Tsin Wong (The Chinese University of Hong Kong) â€‹(*joint first authors)

CariGANs: Unpaired Photo-to-Caricature Translation

Kaidi Cao (Tsinghua University and Microsoft Research Asia), Jing Liao (City University of Hong Kong), Lu Yuan (Microsoft Research Asia)

DeepLens: Shallow Depth Of Field From A Single Image

Lijun Wang (Dalian University of Technology), Xiaohui Shen (ByteDance AI Lab), Jianming Zhang, Oliver Wang, Zhe Lin (Adobe Research), Chih-Yao Hsieh, Sarah Kong (Adobe Systems), Huchuan Lu (Dalian University of Technology)

Invertible Grayscale

Menghan Xia (Shenzhen Institutes of Advanced Technology and The Chinese University of Hong Kong), Xueting Liu (The Chinese University of Hong Kong), Tien-Tsin Wong (The Chinese University of Hong Kong and Shenzhen Institutes of Advanced Technology) â€‹

Faces, Faces, Faces

Warp-guided GANs for Single-Photo Facial Animation
Jiahao Geng (Zhejiang University), Tianjia Shao (University of Leeds), Youyi Zheng, Yanlin Weng, Kun Zhou (Zhejiang University)

Practical Dynamic Facial Appearance Modeling and Acquisition
Paulo Gotardo (Disney Research), Jeremy Riviere (Imperial College London), Derek Bradley (Disney Research), Abhijeet Ghosh (Imperial College London)

Stabilized Real-time Face Tracking via a Learned Dynamic Rigidity Prior
Chen Cao, Menglei Chai, Oliver Woodford, Linjie Luo (Snap Inc.)

Deep Incremental Learning for Efficient High-Fidelity Face Tracking
Chenglei Wu, Takaaki Shiratori, Yaser Sheikh (Facebook Reality Labs)

Acquisition, Rendering and Display for Virtual Reality

A System for Acquiring, Processing, and Rendering Panoramic Light Field Stills for Virtual Reality
Ryan S. Overbeck, Daniel Erickson, Daniel Evangelakos, Matt Pharr, Paul Debevec (Google Inc.)

Towards Multifocal Displays with Dense Focal Stacks
Jen-Hao Chang, B. V. K. Vijaya Kumar, Aswin C. Sankaranarayanan (Carnegie Mellon University)

Shading Atlas Streaming
Joerg Mueller, Philip Voglreiter, Mark Dokter, Thomas Neff, Mina Majar, Markus Steinberger, Dieter Schmalstieg (TU Graz)

DeepFocus: Learned Image Synthesis for Computational Displays
Lei Xiao, Anton Kaplanyan, Alexander Fix, Matt Chapman, Douglas Lanman (Facebook Reality Labs)

Advanced SVBRDF

Practical SVBRDF Acquisition of 3D Objects with Unstructured Flash Photography
Giljoo Nam Joo Ho Lee (KAIST), Diego Gutierrez (Universidad de Zaragoza), Min H. Kim (KAIST)

Simultaneous Acquisition of Polarimetric SVBRDF and Normals
Seung-Hwan Baek, Daniel S. Jeon (KAIST), Xin Tong (Microsoft Research Asia), Min H. Kim (KAIST)

Learning to Reconstruct Shape and Spatially-Varying Reflectance from a Single Image
Zhengqin Li, Zexiang Xu, Ravi Ramamoorthi (University of California, San Diego), Kalyan Sunkavalli (Adobe Research), Manmohan Chandraker (University of California, San Diego)

Lowâ€“Level Imaging

Image Smoothing via Unsupervised Learning
Qingnan Fan (Shandong University), Jiaolong Yang, David Wipf (Microsoft Research Asia), Baoquan Chen (Shandong University), Xin Tong (Microsoft Research Asia)

Image Super-Resolution via Deterministic-Stochastic Synthesis and Local Statistical Rectification
Weifeng Ge, Bingchen Gong, Yizhou Yu (The University of Hong Kong)

Two-Stage Sketch Colorization
LvMin Zhang* (Soochow University), Chengze Li* Tien-Tsin Wong (The Chinese University of Hong Kong), Yi Ji, ChunPing Liu (Soochow University) â€‹(*joint first authors)

Efficient palette-based decomposition and recoloring of images via RGBXY-space geometry
Jianchao Tan, Jose Echevarria, Yotam Gingold (George Mason University)