UNIVERSITY OF SOUTH CAROLINA

CSCE 590 INTRODUCTION TO IMAGE PROCESSING

Motion and Optical Flow

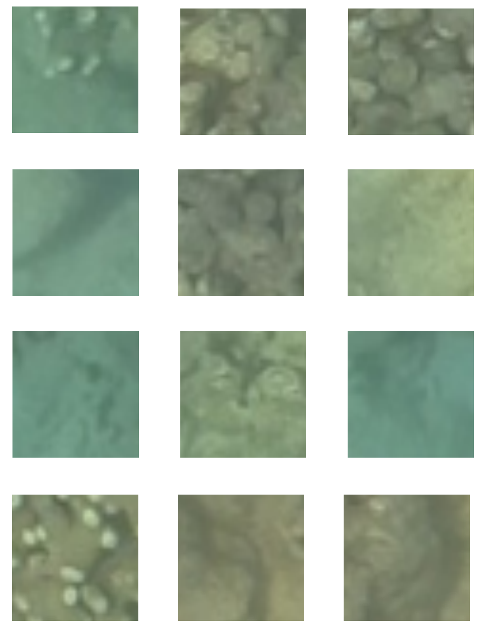
Ioannis Rekleitis

Correspondence

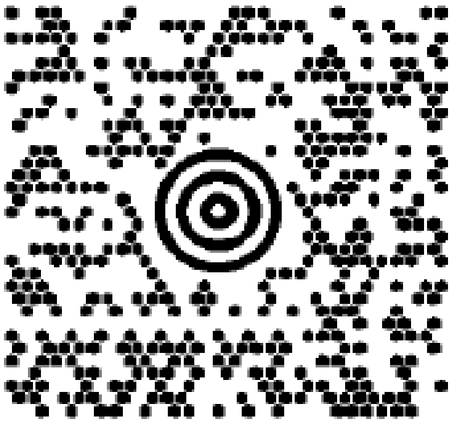
From I_1

?

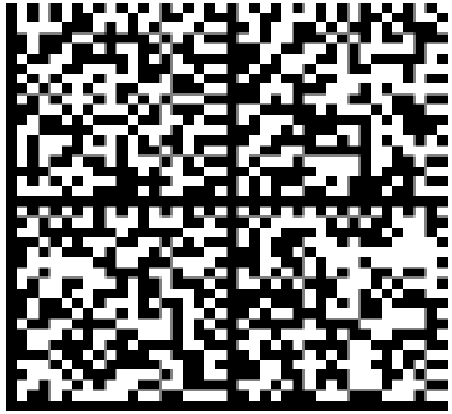
From I_2



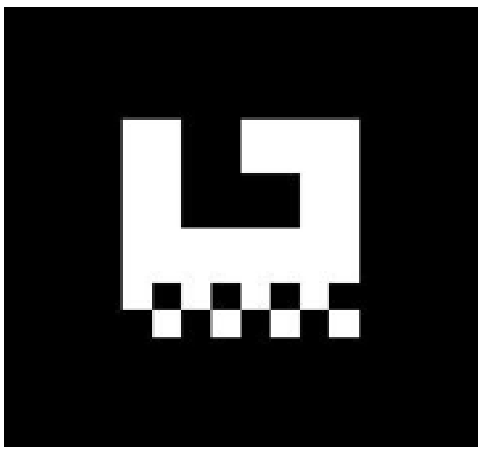
Fiduciary Markers/Fiducial



(a) MaxiCode

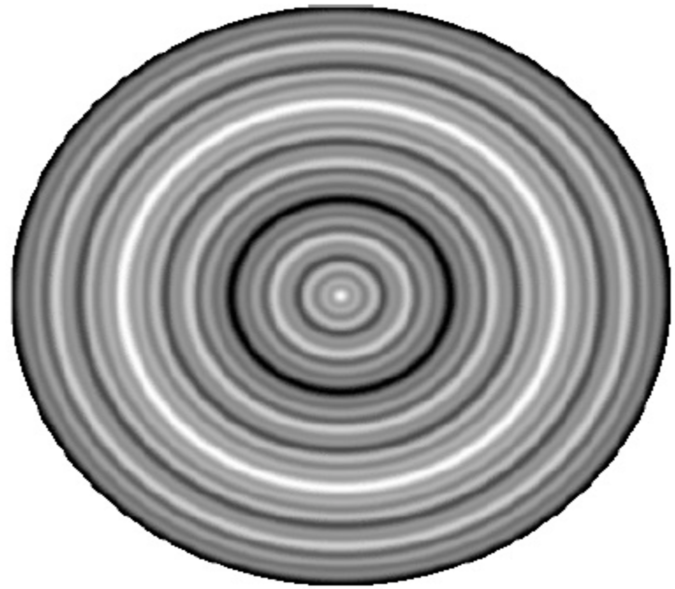


(b) DataMatrixSymbol



(c) ARToolkit

(d) ARTag

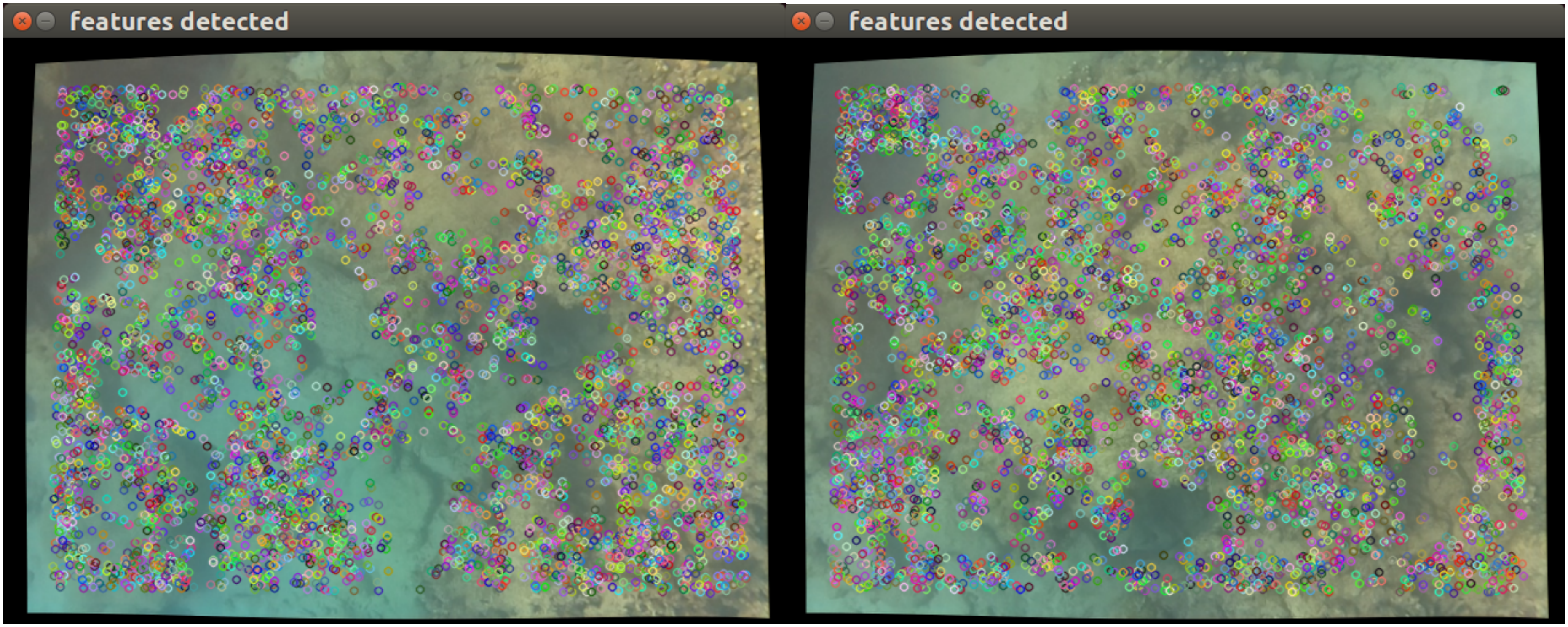


Fourier Tag

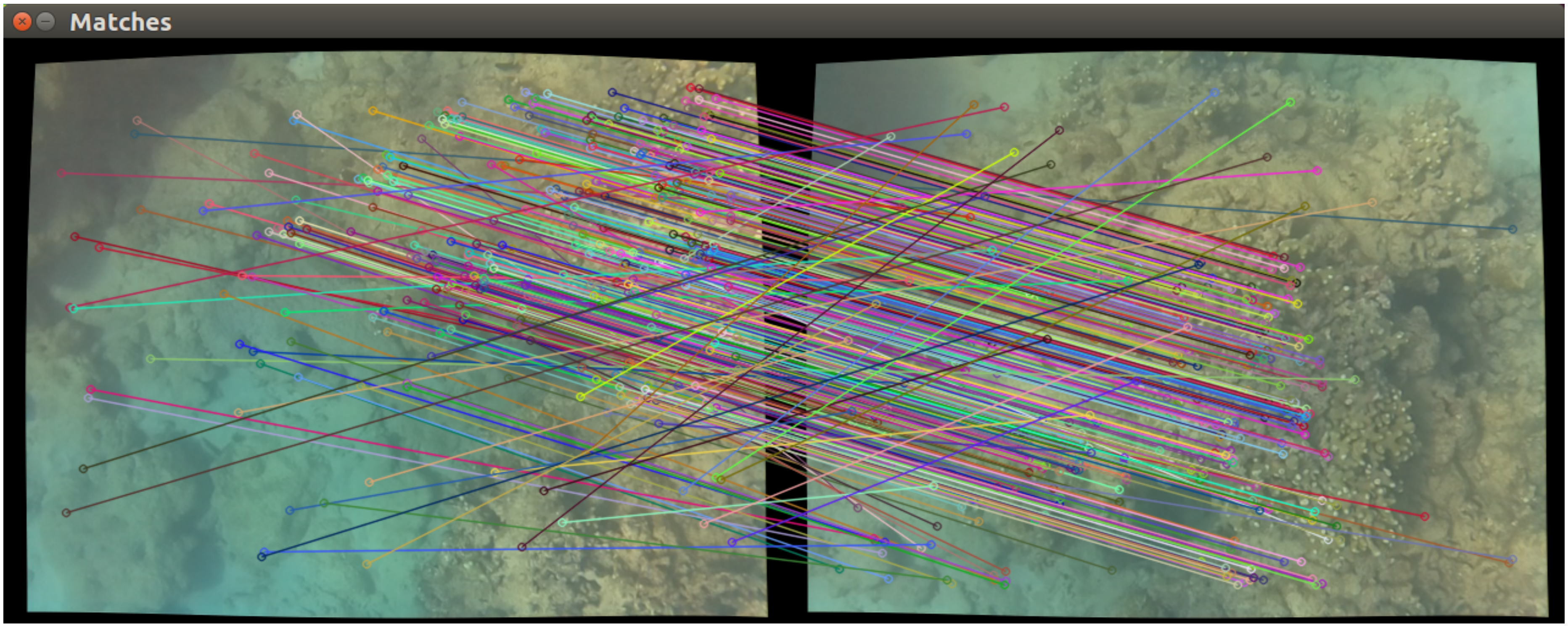
Good Feature

- Invariant to transformations
- Unique
- Efficient to compute
- Good precision and high recall
- Several Alternatives:
  - Harris Corners (OpenCV)
  - SURF (OpenCV)
  - SIFT
  - ORB
  - etc

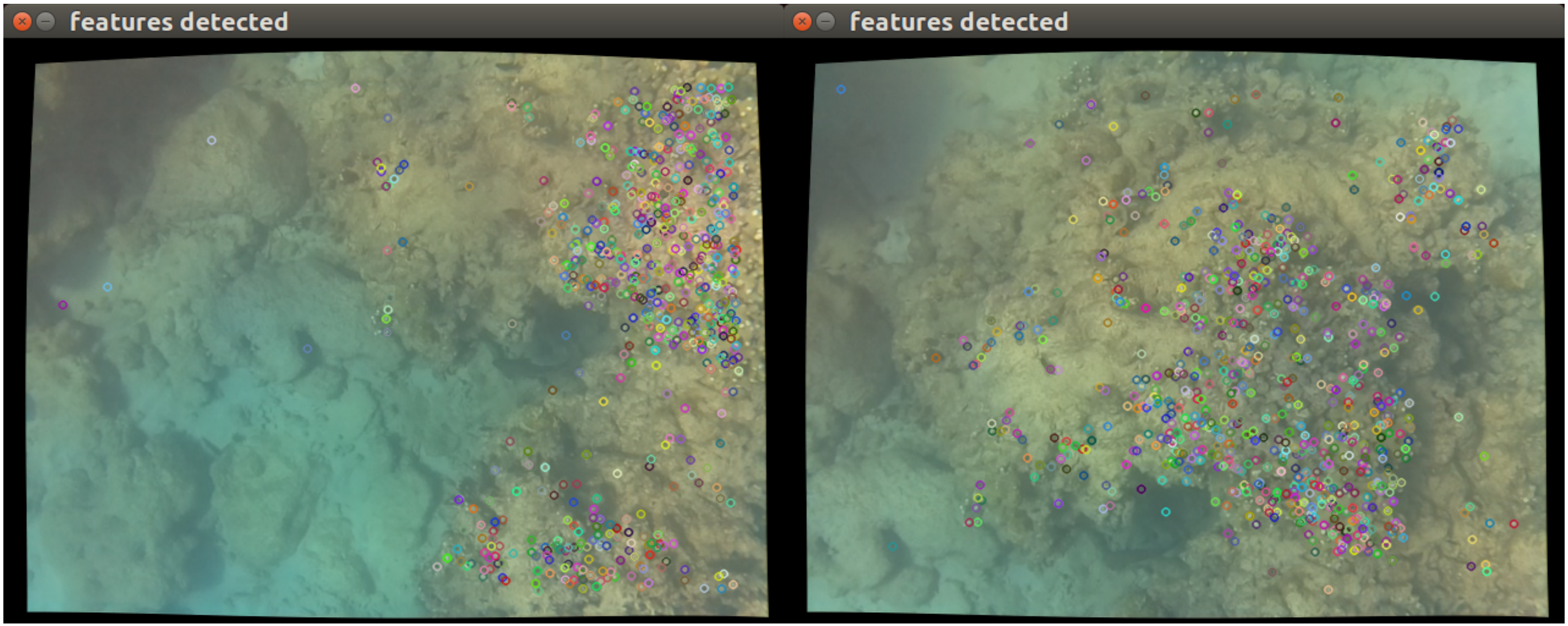
Harris Corners



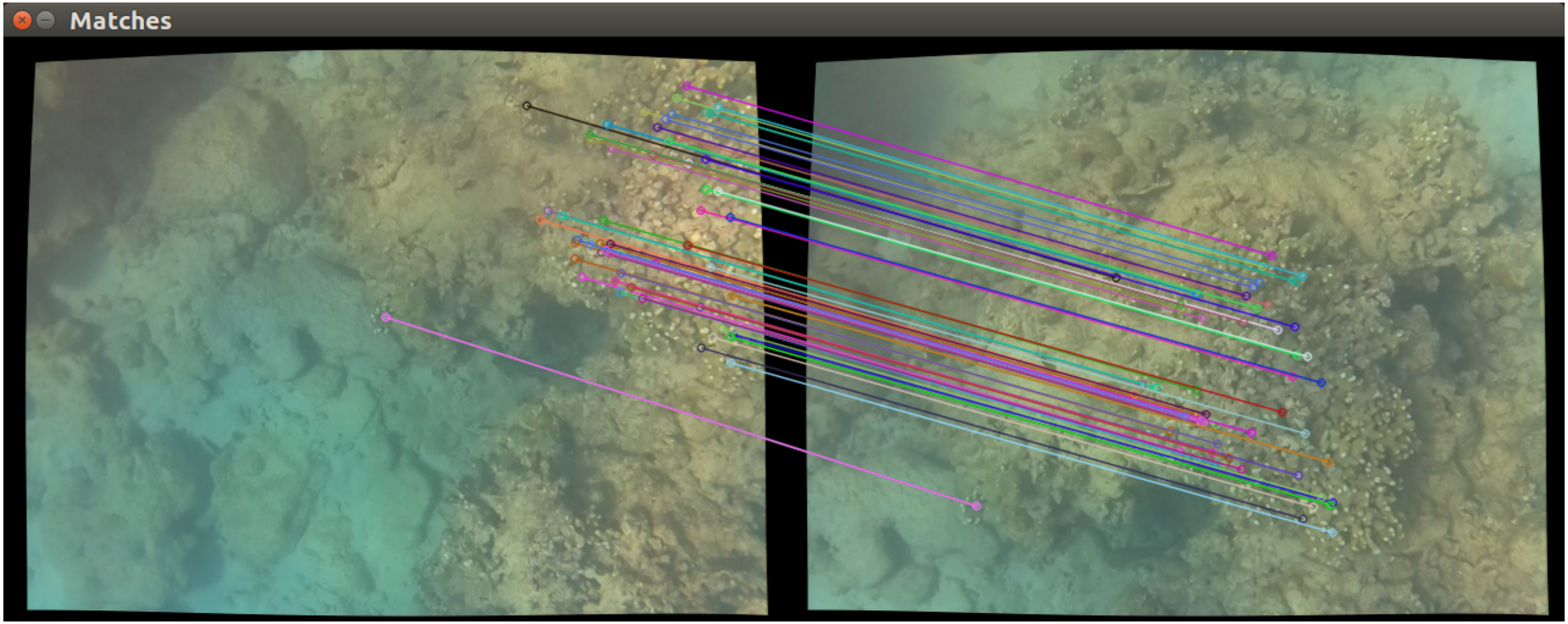
Harris Corners



SIFT

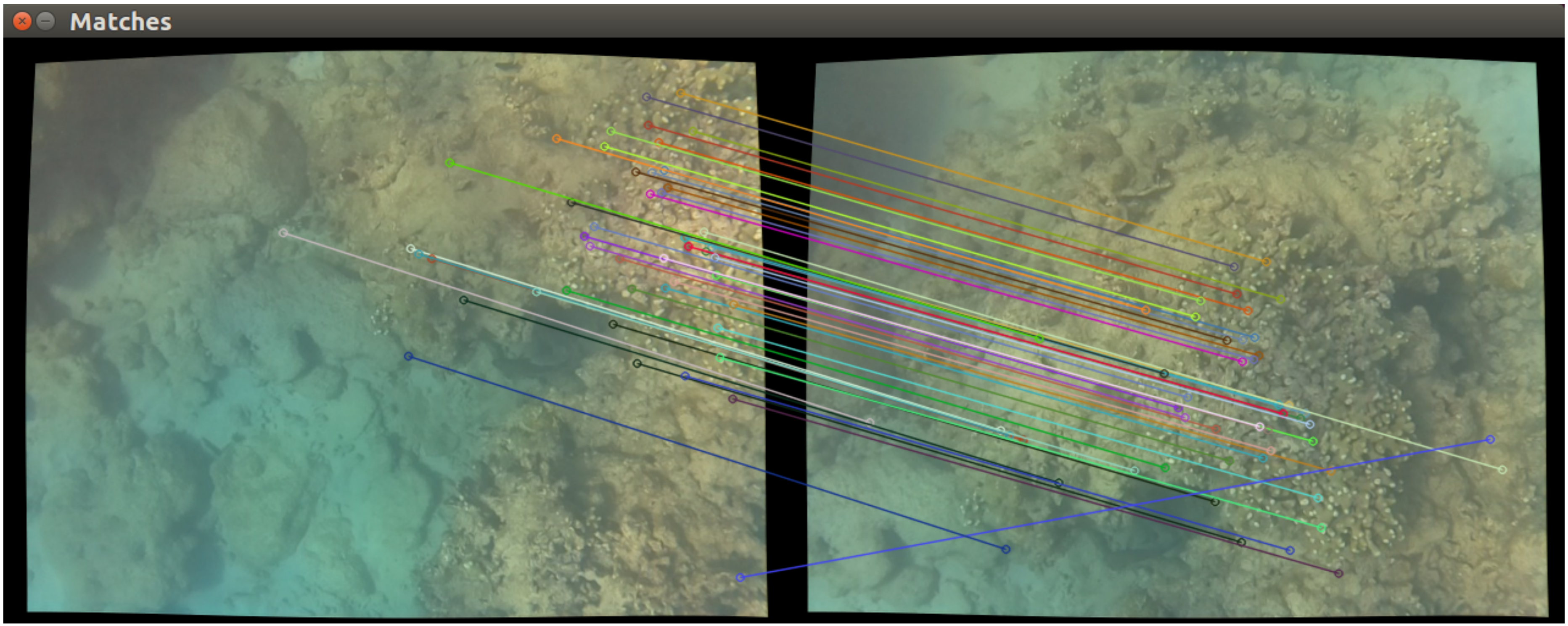


SIFT

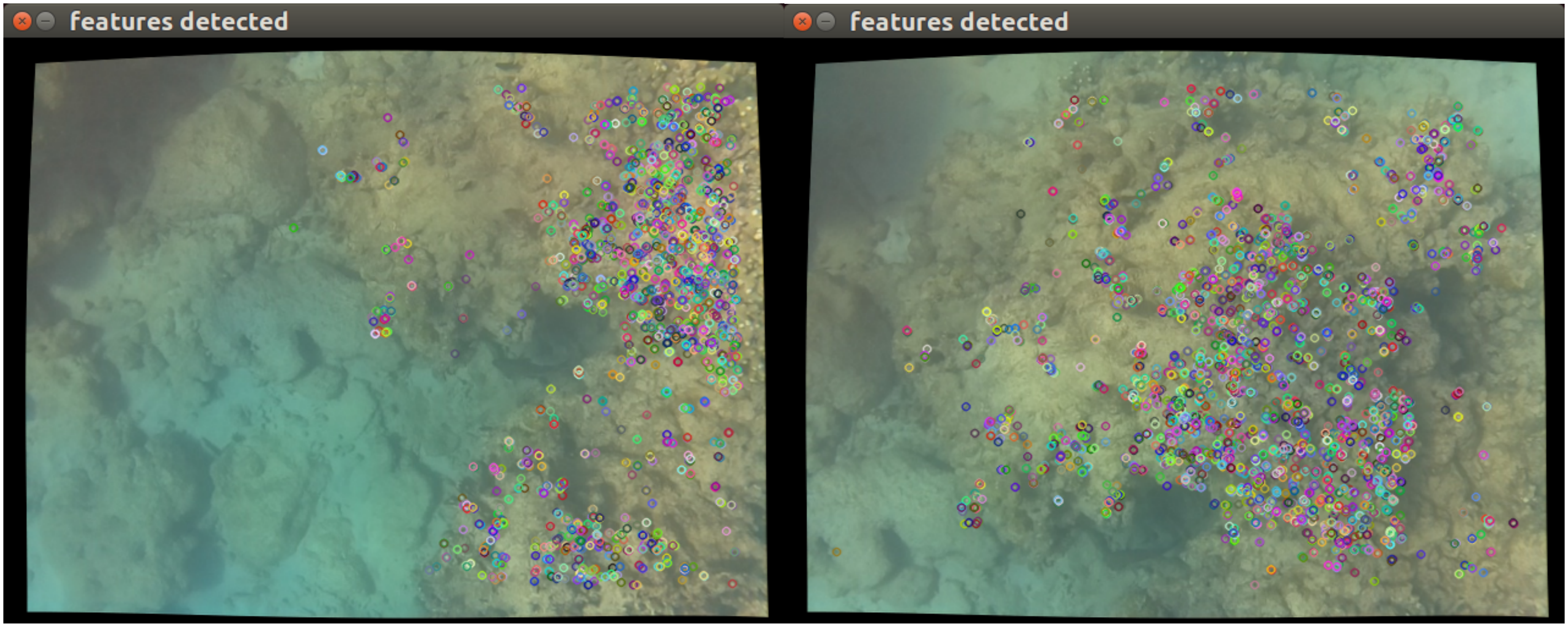


SURF

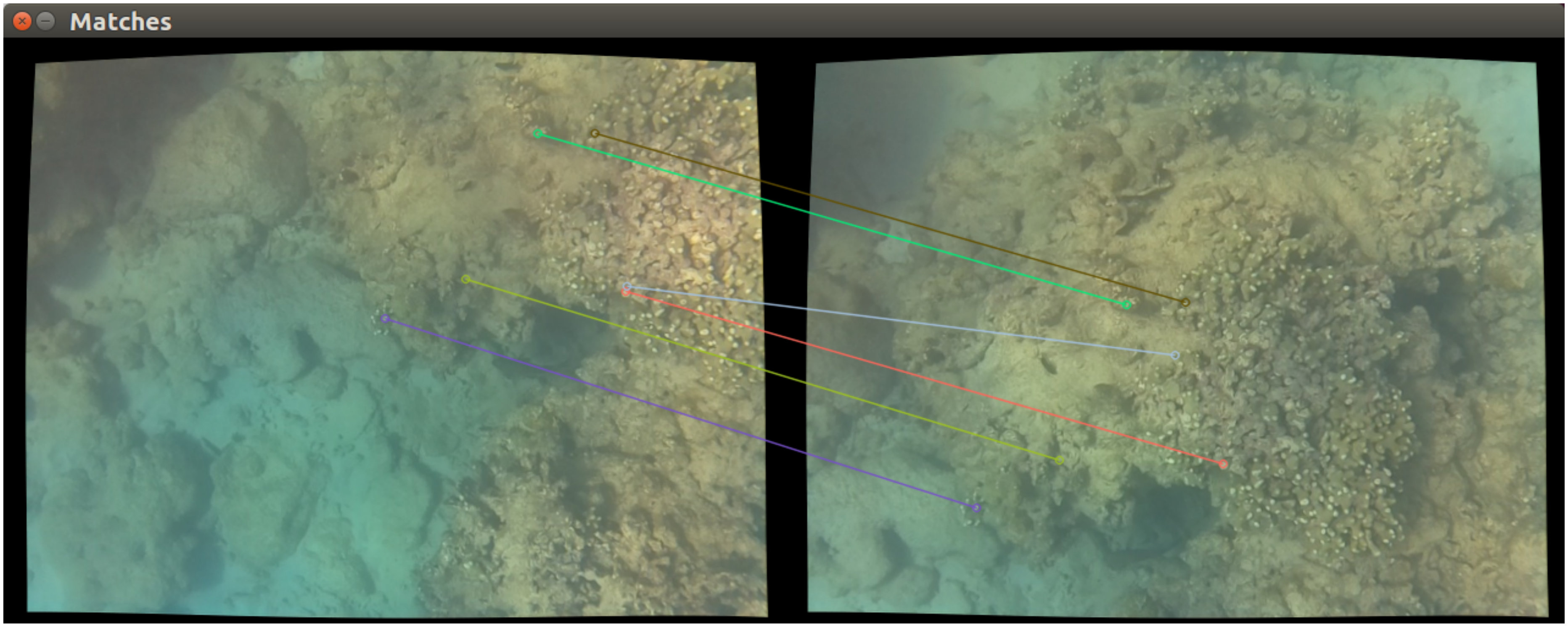
SURF



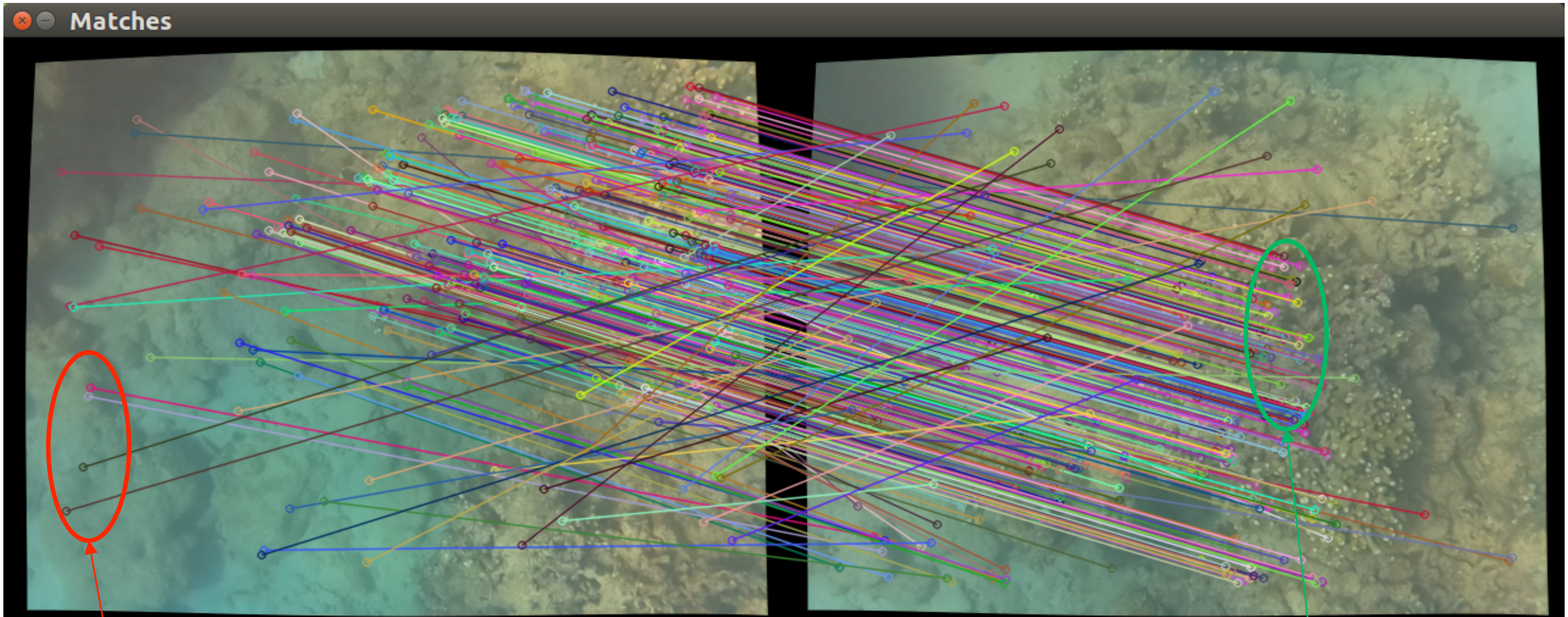
ORB



ORB



Outliers



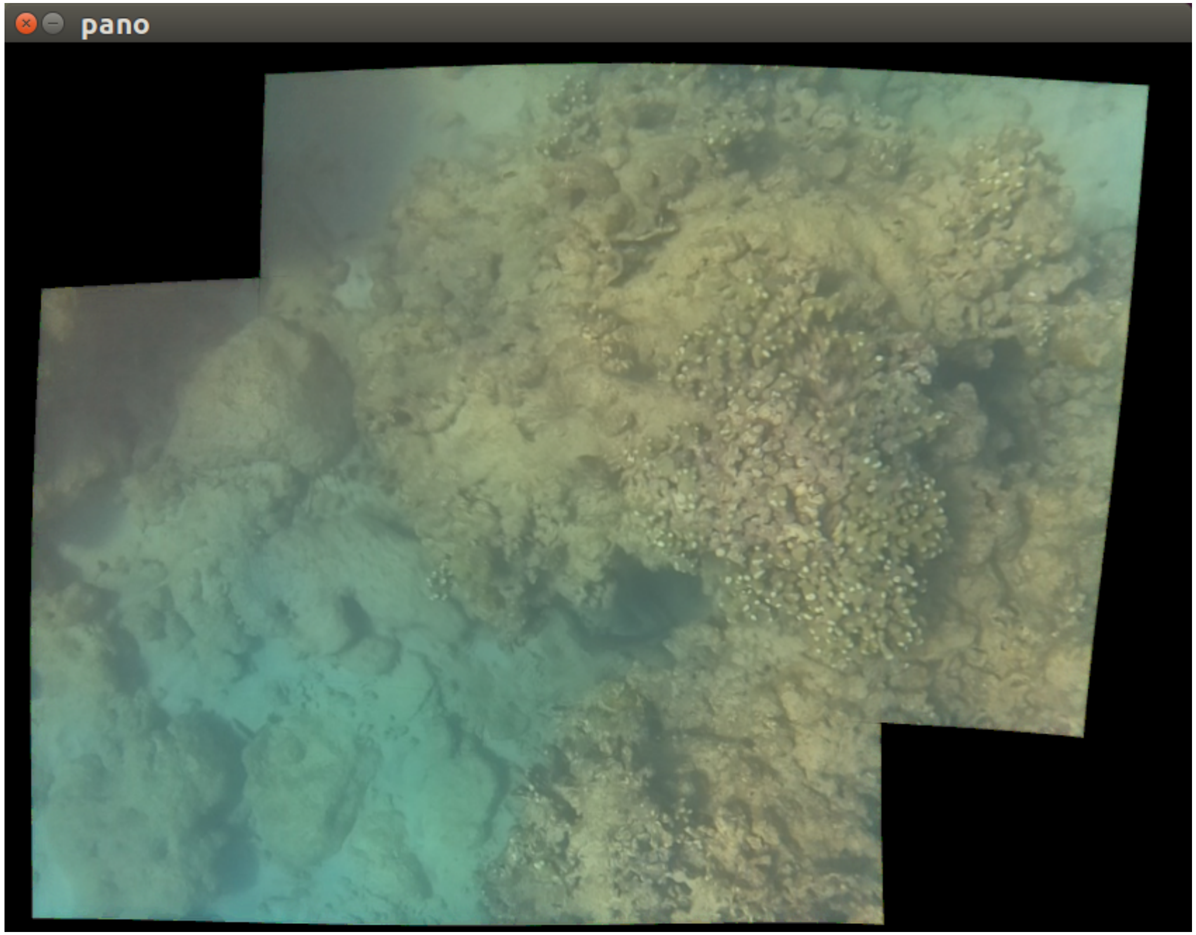
Outliers

Inliers

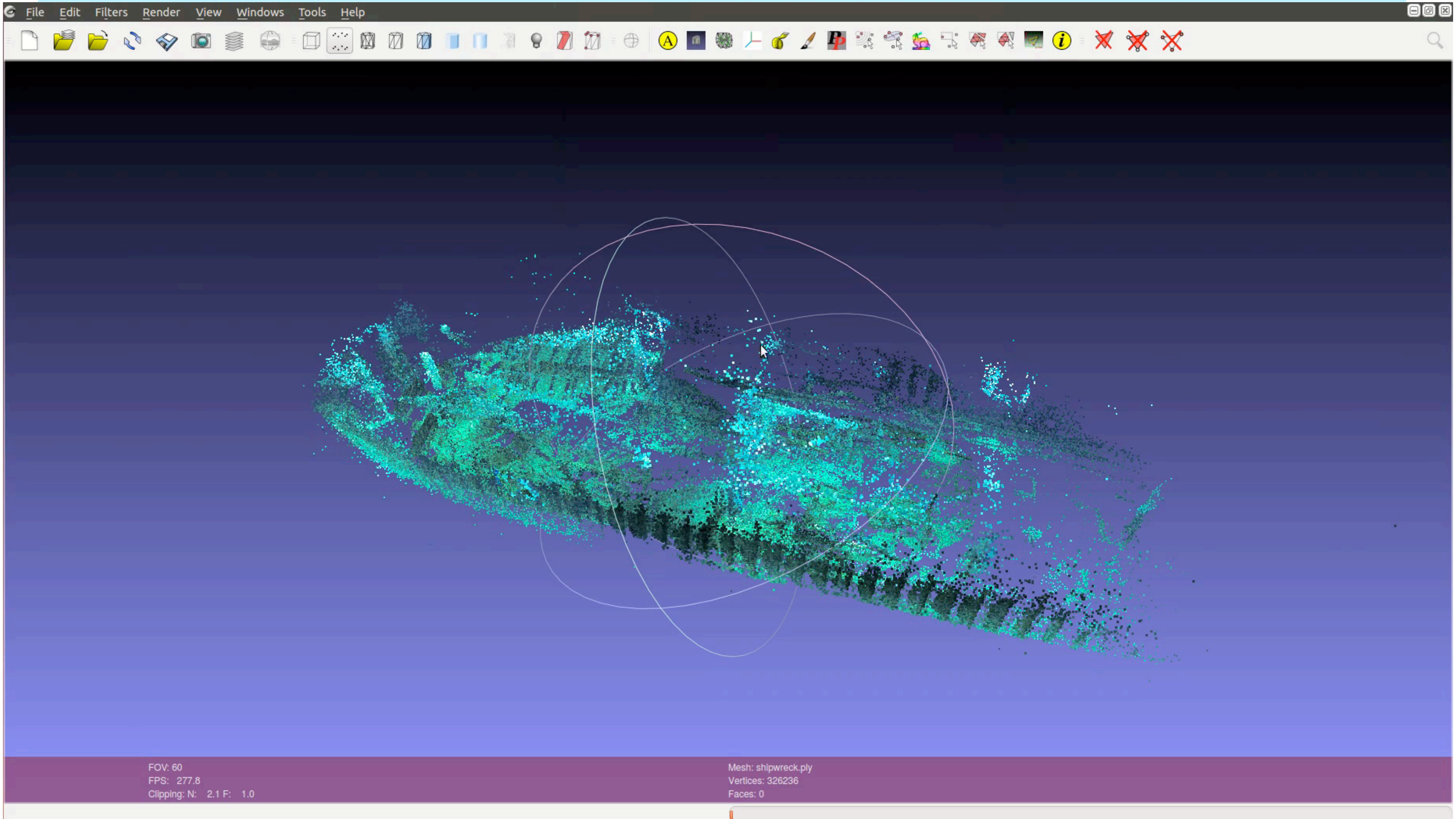
RANSAC

- See Visual Odometry Tutorial Presentation

Mosaic



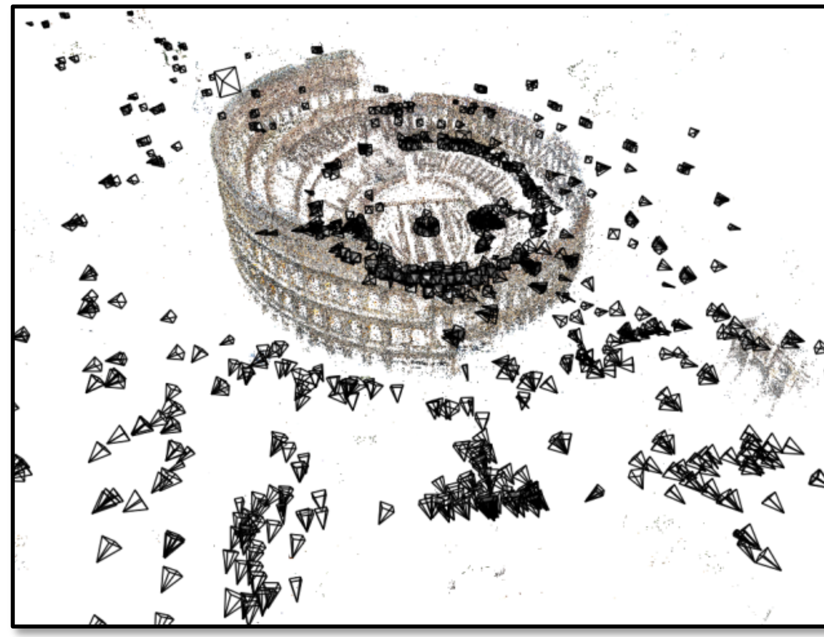
3D Sparse reconstruction



3D Sparse reconstruction

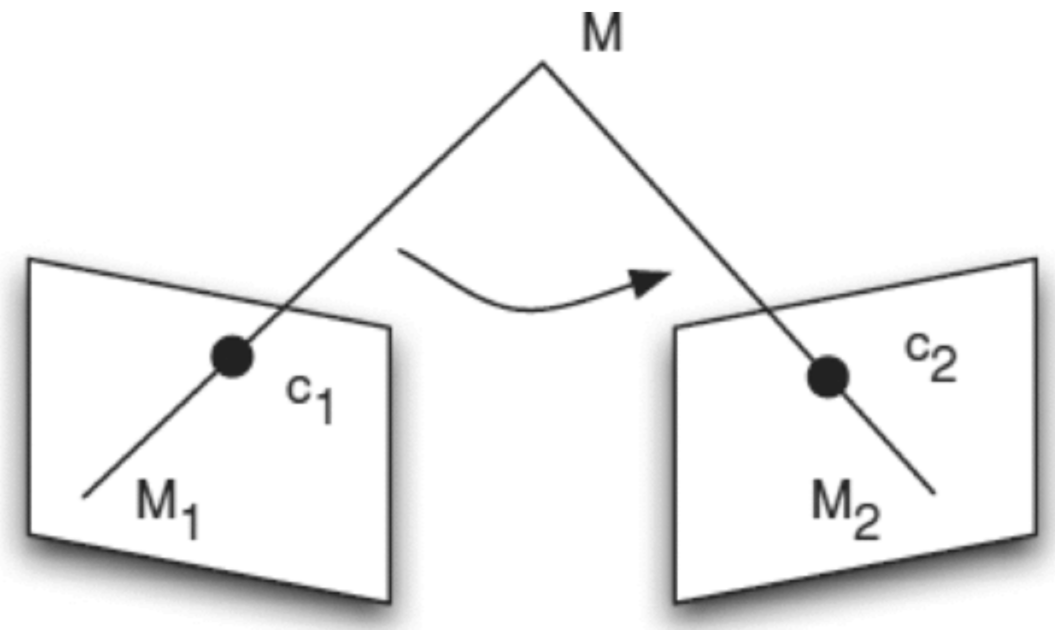
Source: https://grail.cs.washington.edu/rome/

Internet Photos ("Colosseum")



Reconstructed 3D cameras and points

Egomotion



$$C_1 M_1 (T \times R C_2 M_2) = 0$$

Visual Odometry/Structure from Motion

Image stream → Feature detection → Feature matching (tracking) → Motion estimation → Optimization

Optical Flow

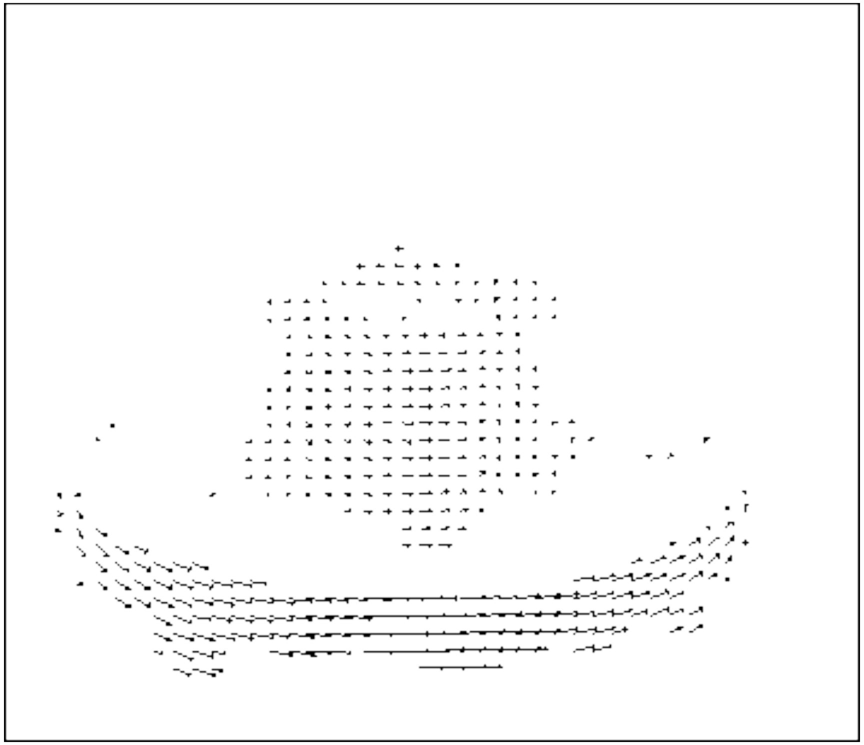
- Definition:
  - *the pattern of apparent motion of objects, surfaces, and edges in a visual scene caused by the relative motion between an observer (an eye or a camera) and the scene.*

Difference between Optical Flow and Scene Motion

- Optical flow: change in the image (2D)
- Scene Motion: change in the scene (3D)

Optical Flow Field

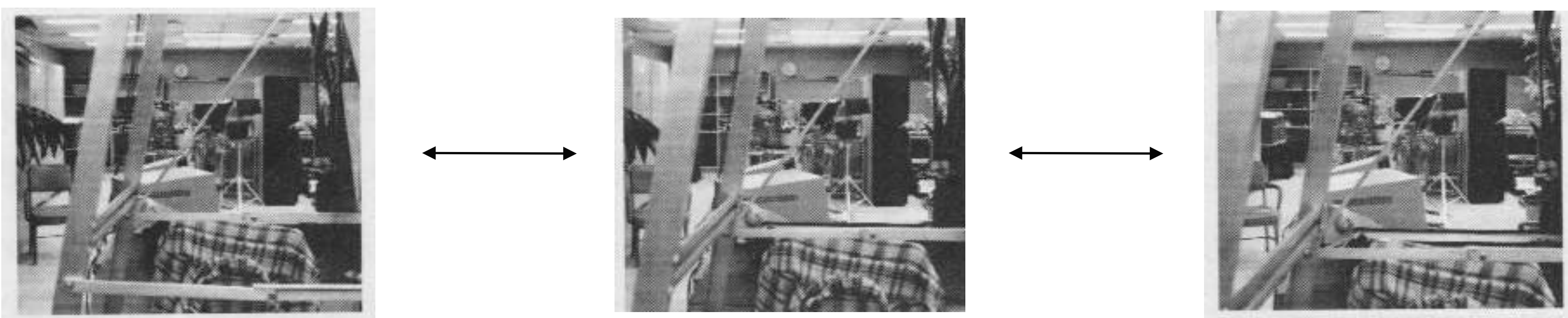




Optical flow

Information about *image motion* rather than the *scene.*
This is a classic ***reconstruction*** *problem.*

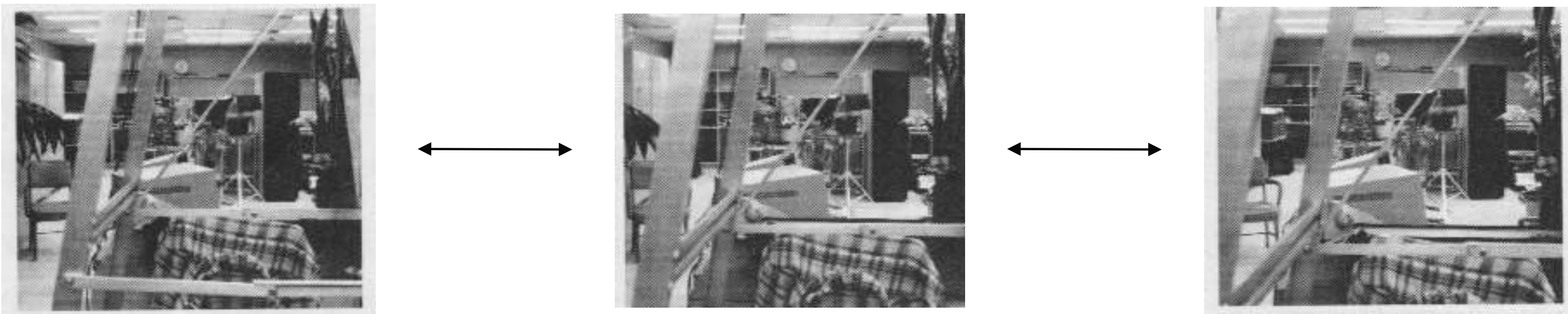
This next step might be to use the image motion to infer scene motion, robot motion or 3D layout.



time sequence of images

Optical flow

Information about *scene motion* rather than the *scene.*

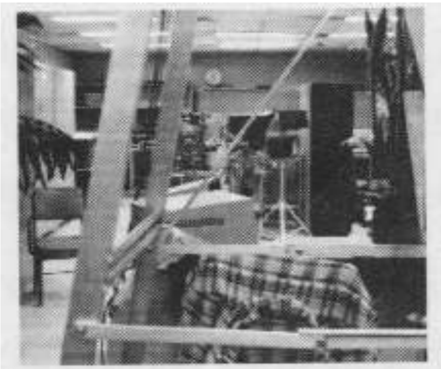


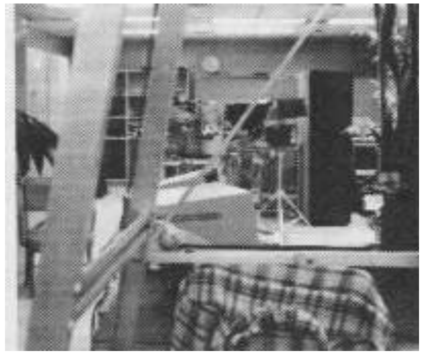
an “image cube”

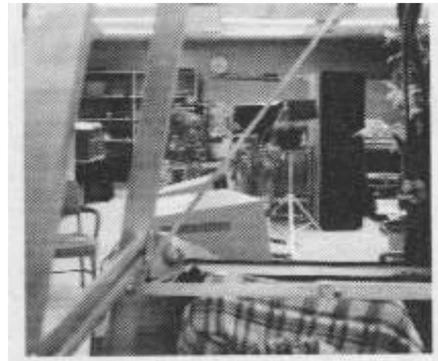
$I(x,y,t)$

Optical flow

Information about *scene motion* rather than the *scene.*







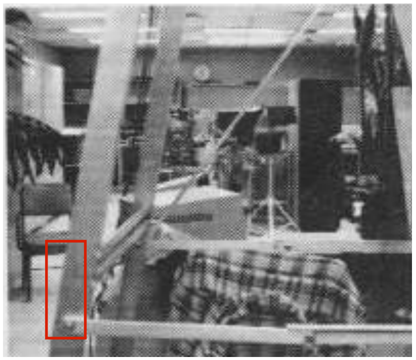
optical flow

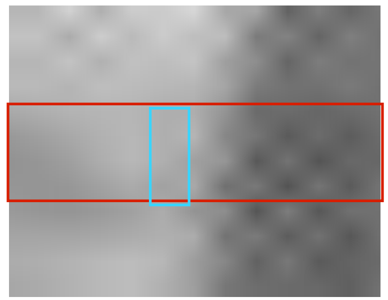
How ?

Optical Flow

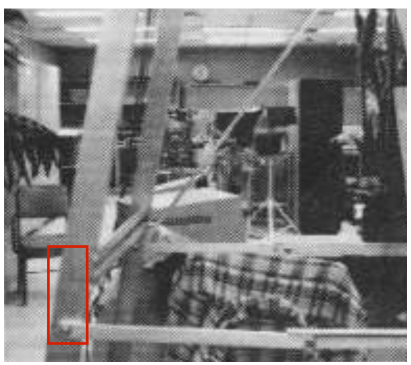
- By measuring the direction that intensities are moving...

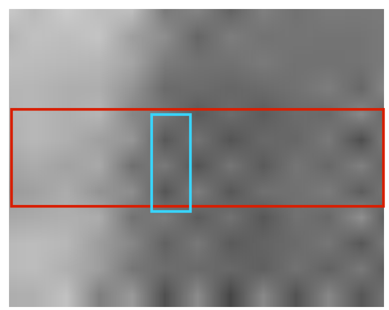
$I(x,y,t)$





99	90	90	70	40
95	90	70	40	40
90	90	70	40	40
90	90	70	40	40
90	70	50	40	30





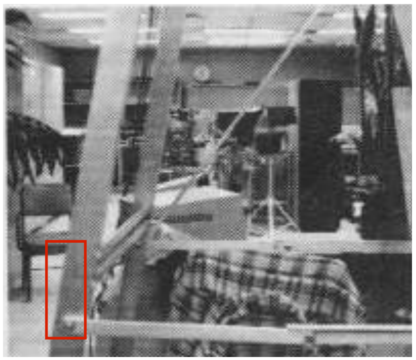
90	90	70	40	25
90	70	40	40	25
90	70	40	40	25
90	70	40	40	20
70	50	40	30	15

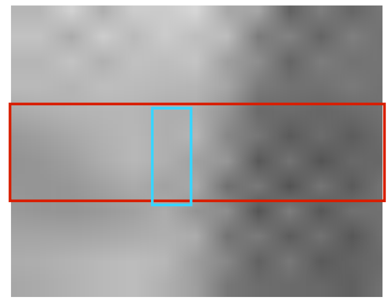
- We can estimate things...

Optical Flow

By measuring the direction that intensities are moving...

I(x,y,t)



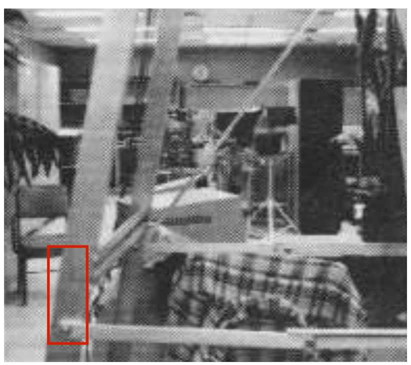


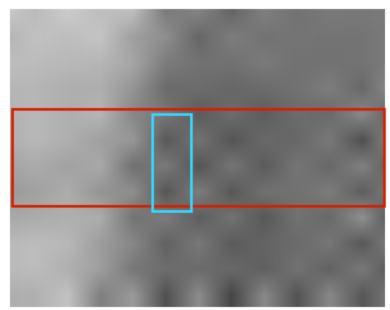
I(x,y,0)

99	90	90	70	40
95	90	70	40	40
90	90	70	40	40
90	90	70	40	40
90	70	50	40	30

I(0,0,0)

I(2,-1,0)





I(x,y,1)

90	90	70	40	25
90	70	40	40	25
90	70	40	40	25
90	70	40	40	20
70	50	40	30	15

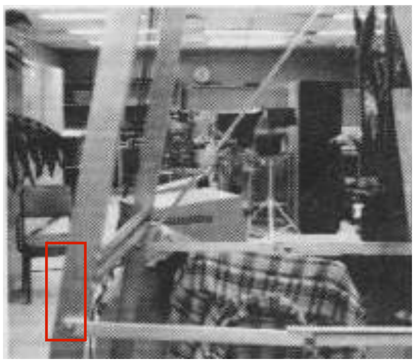
I(0,0,1)

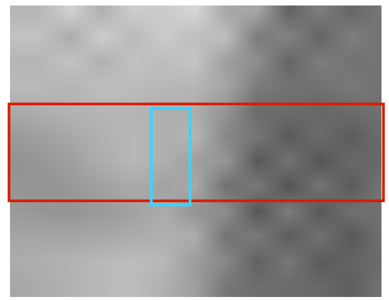
We can estimate things ...

$$\frac{dI}{dx} = I_x \quad \text{at } (0,0,0)$$

Optical Flow

By measuring the direction that intensities are moving...





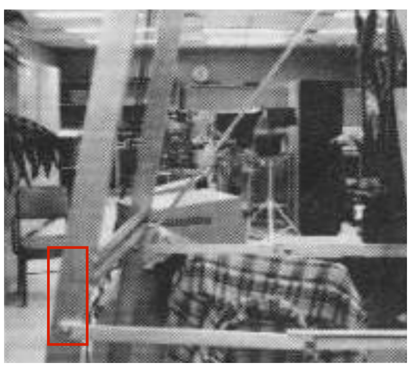
I(x,y,t)

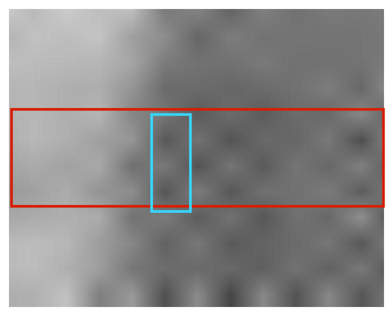
I(x,y,0)

99	90	90	70	40
95	90	70	40	40
90	90	**70**	40	40
90	90	70	40	**40**
90	70	50	40	30

I(0,0,0)

I(2,-1,0)





I(x,y,1)

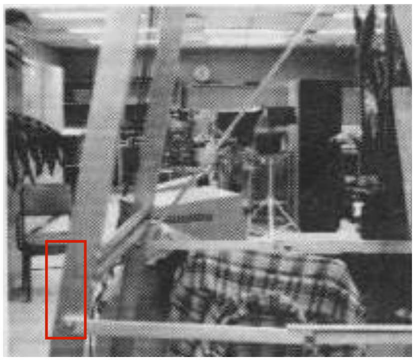
90	90	70	40	25
90	70	40	40	25
90	70	**40**	40	25
90	70	40	40	20
70	50	40	30	15

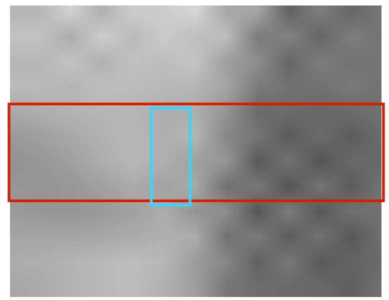
I(0,0,1)

We can estimate things like $\frac{dI}{dx} = I_x$ at (0,0,0) $= \frac{\Delta I}{\Delta x} = \frac{I(1,0,0) - I(0,0,0)}{1 - 0} = -30$

Optical Flow

By measuring the direction that intensities are moving...





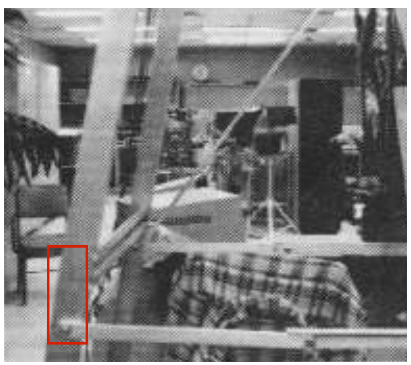
$I(x,y,t)$

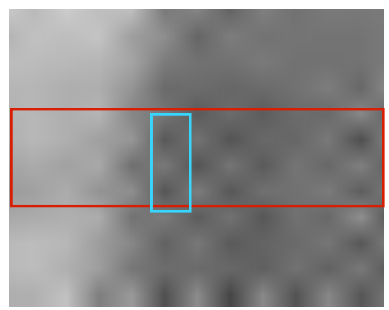
$I(x,y,0)$

99	90	90	70	40
95	90	70	40	40
90	90	**70**	40	40
90	90	70	40	**40**
90	70	50	40	30

$I(0,0,0)$

$I(2,-1,0)$





$I(x,y,1)$

90	90	70	40	25
90	70	40	40	25
90	70	**40**	40	25
90	70	40	40	20
70	50	40	30	15

$I(0,0,1)$

We can estimate things like $\frac{dI}{dx} = I_x$ $\frac{dI}{dy} = I_y$ $\frac{dI}{dt} = I_t$ so...

Measuring Optical Flow

Let I(x,y,t) be the sequence of images.

(x,y,t)

Simplest assumption (constant brightness constraint):

$$I(x,y,t) = I(x + dx, y + dy, t + dt)$$

99	90	90	70	40
95	90	70	40	40
90	90	70	40	40
90	90	70	40	40
90	70	50	40	30

Measuring Optical Flow

Let I(x,y,t) be the sequence of images.

(x,y,t)

Simplest assumption (constant brightness constraint):

99	90	90	70	40
95	90	70	40	40
90	90	70	40	40
90	90	70	40	40
90	70	50	40	30

$$I(x,y,t) = I(x + dx, y + dy, t + dt)$$

Reminder: $f(x + dx) = f(x) + f'(x)\, dx + f''(x)\, dx^2 / 2 + \ldots$

Measuring Optical Flow

Let $I(x,y,t)$ be the sequence of images.

Simplest assumption (constant brightness constraint):

$$I(x,y,t) = I(x + dx, y + dy, t + dt)$$

(x,y,t)

99	90	90	70	40
95	90	70	40	40
90	90	70	40	40
90	90	70	40	40
90	70	50	40	30

Reminder: $f(x + dx) = f(x) + f'(x)\,dx + f''(x)\,dx^2/2 + \ldots$

$$I(x,y,t) = I(x,y,t) + I_x\,dx + I_y\,dy + I_t\,dt + \text{2nd deriv. + higher}$$

Measuring Optical Flow

Let $I(x,y,t)$ be the sequence of images.

(x,y,t)

99	90	90	70	40
95	90	70	40	40
90	90	70	40	40
90	90	70	40	40
90	70	50	40	30

Simplest assumption (constant brightness constraint):

$$I(x,y,t) = I(x + dx, y + dy, t + dt)$$

Reminder: $f(x + dx) = f(x) + f'(x)\,dx + f''(x)\,dx^2/2 + \ldots$

$$I(x,y,t) = I(x,y,t) + I_x\,dx + I_y\,dy + I_t\,dt + \underbrace{\text{2nd deriv. + higher}}_{\text{ignore these terms}}$$

$$0 = I_x\,dx + I_y\,dy + I_t\,dt$$

Measuring Optical Flow

Let $I(x,y,t)$ be the sequence of images.

Simplest assumption (constant brightness constraint):

$$I(x,y,t) = I(x + dx, y + dy, t + dt)$$

(x,y,t)

99	90	90	70	40
95	90	70	40	40
90	90	70	40	40
90	90	70	40	40
90	70	50	40	30

Reminder: $f(x + dx) = f(x) + f'(x)\, dx + f''(x)\, dx^2 /2 + \ldots$

$$I(x,y,t) = I(x,y,t) + I_x\, dx + I_y\, dy + I_t\, dt + \underbrace{\text{2nd deriv. + higher}}_{\text{ignore these terms}}$$

$$0 = I_x\, dx + I_y\, dy + I_t\, dt$$

$$-I_t = I_x \frac{dx}{dt} + I_y \frac{dy}{dt}$$

intensity-flow equation

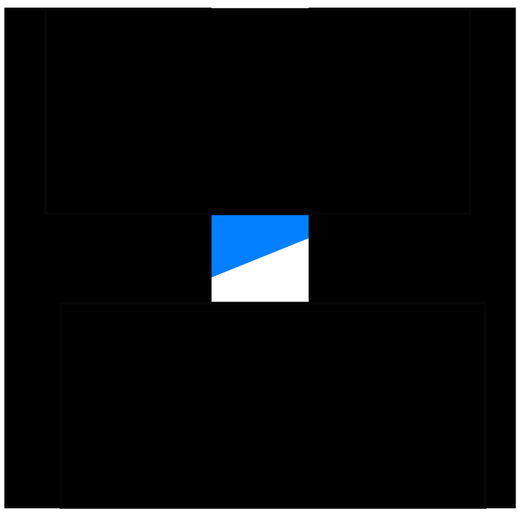
good and bad...

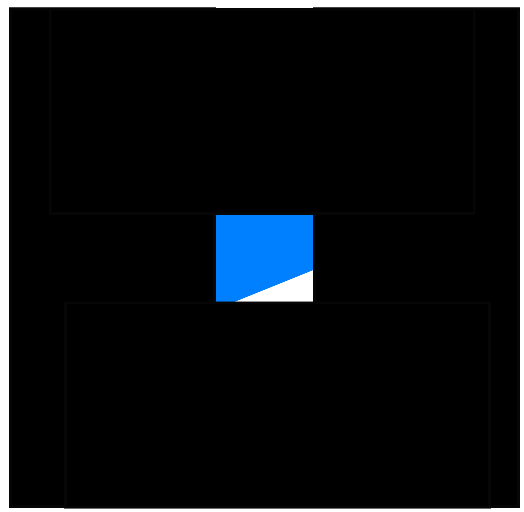
The "aperture" problem

$$-I_t = I_x \frac{dx}{dt} + I_y \frac{dy}{dt}$$

- The intensity-flow equation provides only one constraint on *two* variables (x-motion and y-motion)

$\longrightarrow$ It is only possible to find optical flow in one direction...





The "aperture" problem

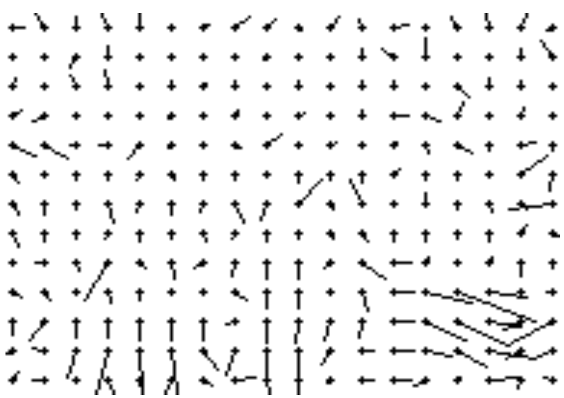
→ It is only possible to find optical flow in one direction...

*at any **single** point in the image !*

img1

img2

raw optical flow



smoothed for ten iterations

Smoothing can be done by incorporating neighboring points' information.

Observations & Warnings

- Assume the scene itself is static.
- Find matching chunks in the images.
- An instance of *correspondence*.

BUT

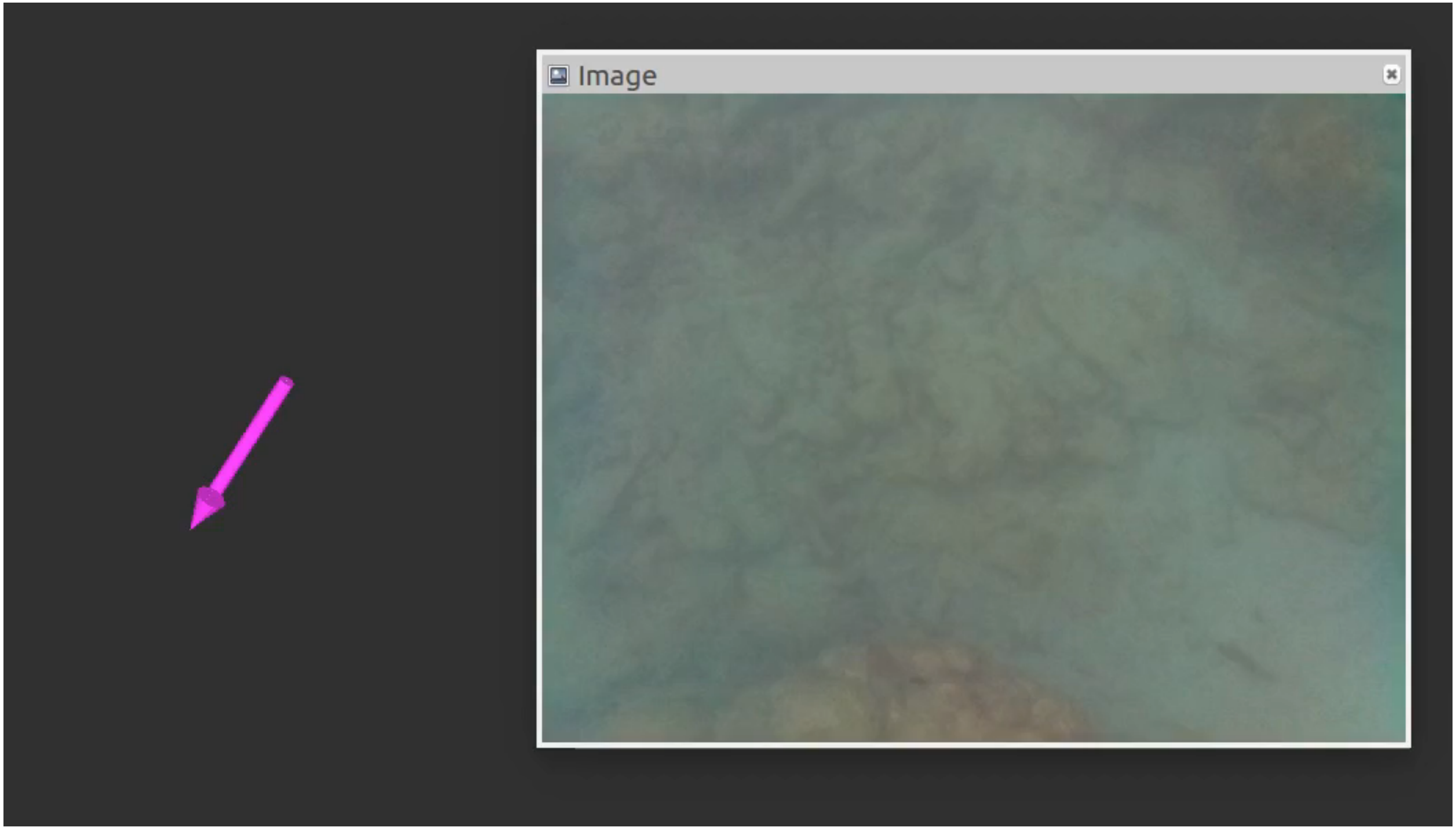
- World really isn't static.
- Lightning might change even in a static scene.

Features vs Optical Flow

- Feature-based methods
  - Detect features (corners, textured areas), extract descriptors, and track them
  - Sparse motion fields, but possibly robust tracking
  - Suitable especially when image motion is large (10s of pixels)
- Direct methods (optical flow)
  - Directly recover image motion from spatio-temporal image brightness variations
  - Global motion parameters directly recovered without an intermediate feature motion calculation
  - Dense motion fields, but more sensitive to appearance variations
  - Suitable for video and when image motion is small (< 10 pixels)

Camera and IMU

From drifter with Raspberry PI Camera and Pololu MinIMU-9 v3 at Barbados 2016 Field Trials

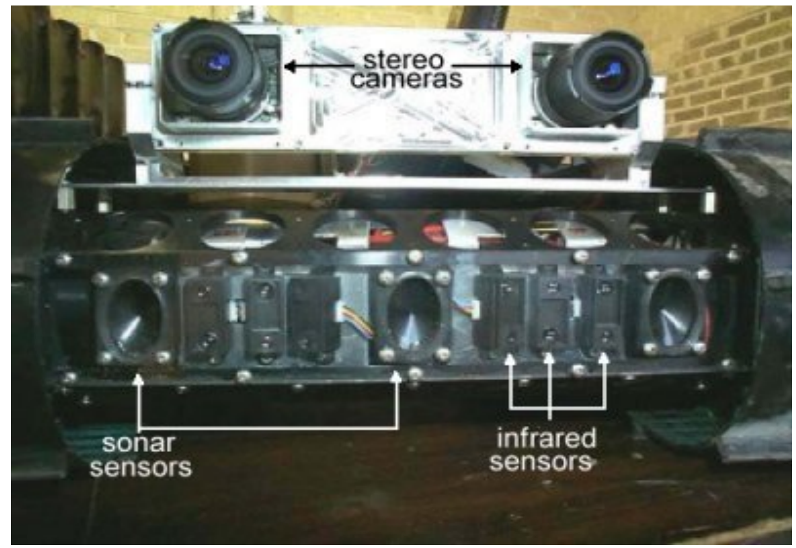


A Vision “solution”

- If interpreting a single image is difficult... What about more ?!

multiple times

multiple cameras



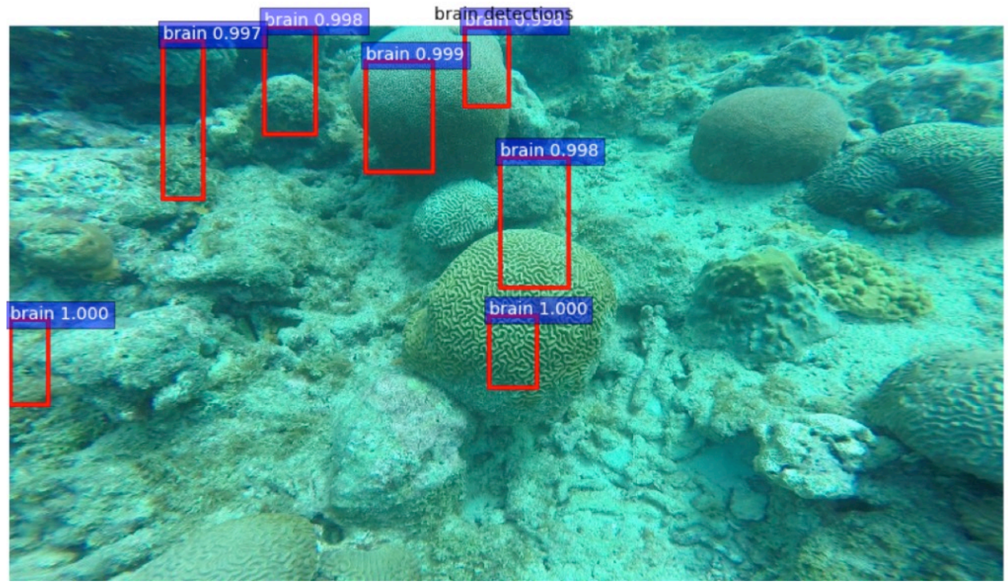
Object recognition

source: http://www.cs.cornell.edu/courses/cs4670/2013fa/

Pedestrian and car detection

Lane detection

From GoPro 3D Hero at Barbados 2015 Field Trial



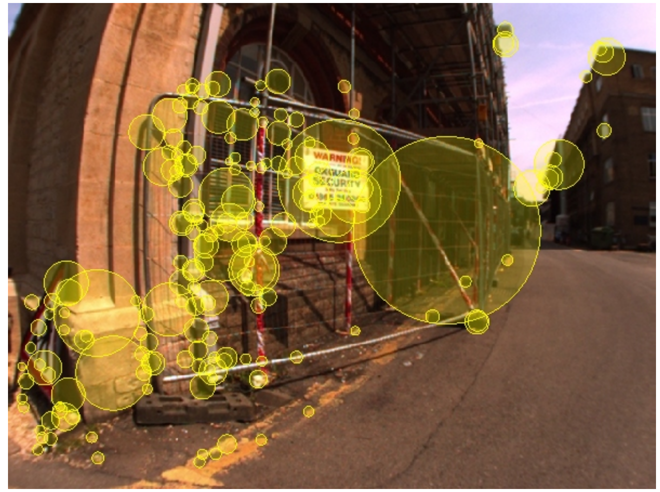
Coral classification

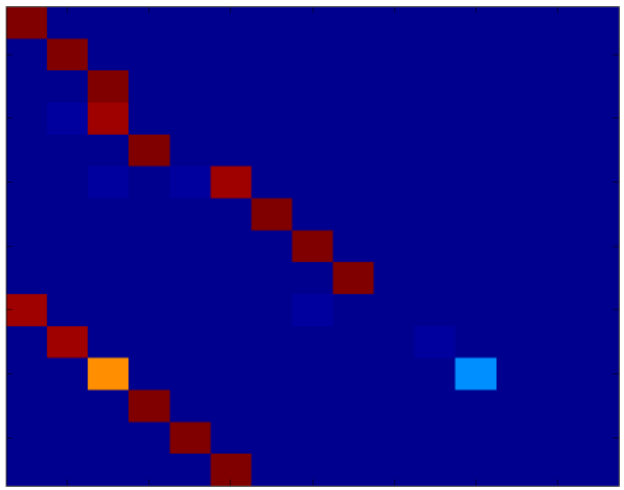
Bag of words

Object → **Bag of ‘words’**

source: http://wikimedia.org

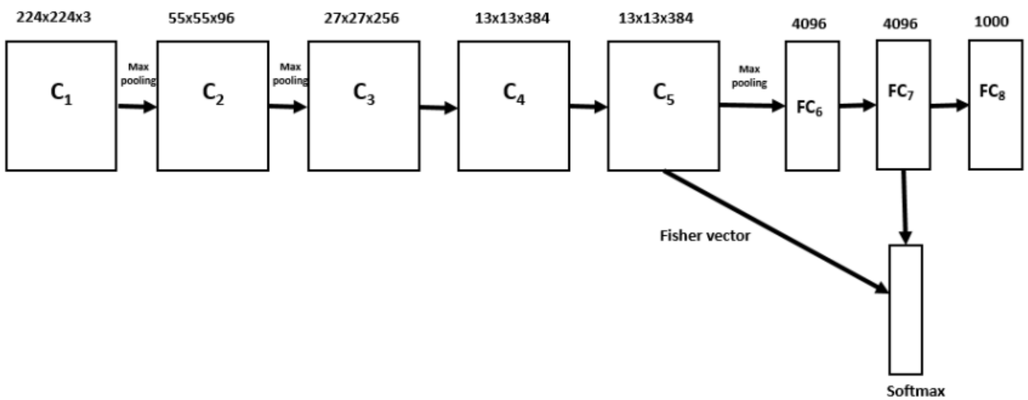
Appearance-based place recognition

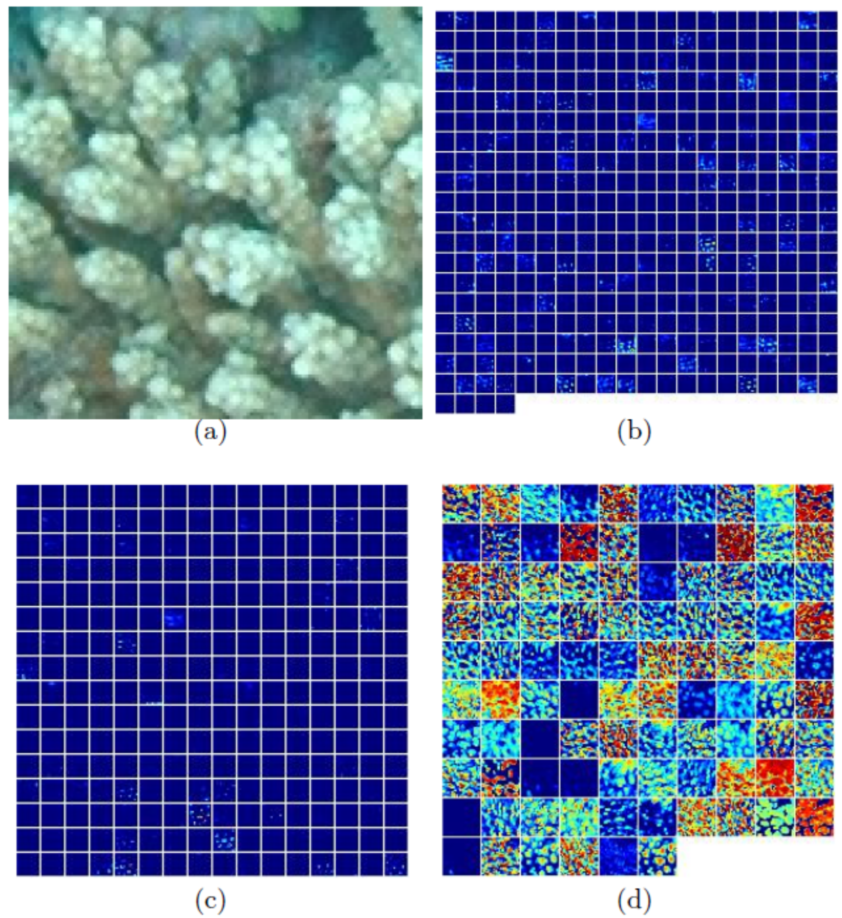




source: http://www.robots.ox.ac.uk/~mjc

Deep learning based classification





Computer Vision Books

- Richard Szeliski, “Computer Vision: Algorithms and Applications”, Springer, 2010
- Richard Hartley and Andrew Zisserman, “Multiple View Geometry in Computer Vision”, Cambridge University Press, 2004
- David Forsyth and Jean Ponce, “Computer Vision: A Modern Approach”, Pearson, 2011

Nice Classes

- Noah Snavely – Introduction to Computer Vision http://www.cs.cornell.edu/courses/cs4670/2013fa/lectures/lectures.html
- Steve Seitz and Rick Szeliski – Computer Vision http://courses.cs.washington.edu/courses/cse576/08sp/

Questions?