Computer Vision Assignment 2

Student Guide / Answers / Feed-back

John Bastian

Australian Institute for Machine Learning
The University of Adelaide

June 2020

3D Reconstruction Overview

- 3D reconstruction involves estimating the 3D position of scene points observed by one (or more) cameras
- image correspondences are used to *triangulate* the 3D point
  - requires knowing the camera geometry, and
  - *accurate matches*
- identifying matching points is difficult
  - assignment 1 illustrated the challenges at block/pixel level
  - ideally, triangulation has *sub-pixel* accuracy!
- Assignment 2: *what is the effect of error in point correspondences?*

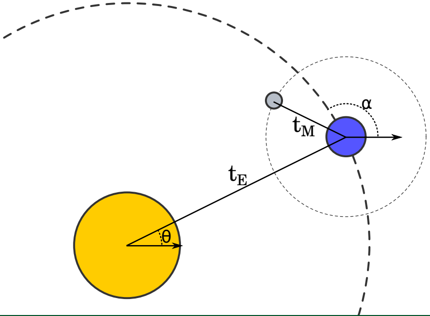
Task 1: focal length

- key goal is to gain familiarity with projection
- change in focal length:
  - equivalent to changing scale
- moving the camera is not the same effect
  - → dolly zoom: changing focal length while moving the camera keeps the foreground at a similar scale while changing the background scale

Dolly zoom: moving the camera ***and*** changing the focal length!

Task 1: transforms

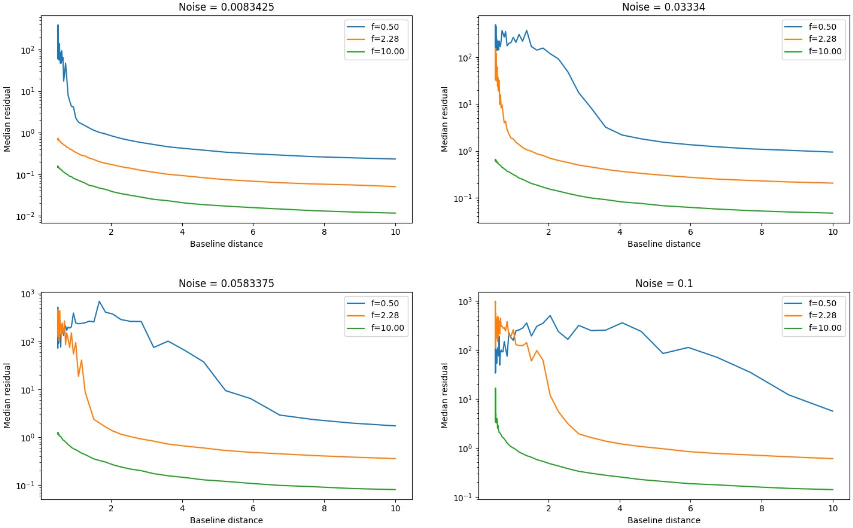
- needed to demonstrate that you could move the camera predictably
  - particularly important for Task 3 which required a compound transform
- compound transformations needed to clearly articulate order
  - "first" depends on your point-of-view;
  - the 'last' term is the 'first' that is applied to the scene!
- aside: you can stack arbitrarily long transforms[1]
  - e.g. the centre of the Moon relative to the Sun is given by $\mathbf{R}_\theta \mathbf{T}_{t_E} \mathbf{R}_{\alpha-\theta} \mathbf{T}_{t_M}$ where $\mathbf{R}$ is the rotation matrix and $\mathbf{T}_x$ is translation by $[x, 0]$



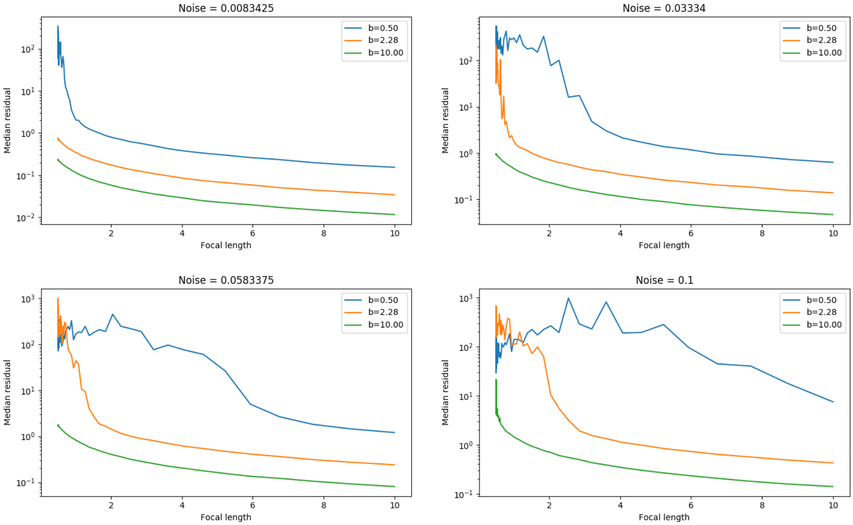
Task 2

- describe how the scene is set up
  - mostly to verify that all points were in front of the cameras
  - (points behind the cameras will obfuscate your results)
  - also keen to see if you recognised that depth variation is important!
- reconstruction error can be measured by Euclidean distance between the GT and triangulated points
  - the average is affected by outliers, so you could also consider the median error!
- generate graphs by varying the camera parameters and amount of noise
  - easiest to automate this rather than use a sparse set of experiments!

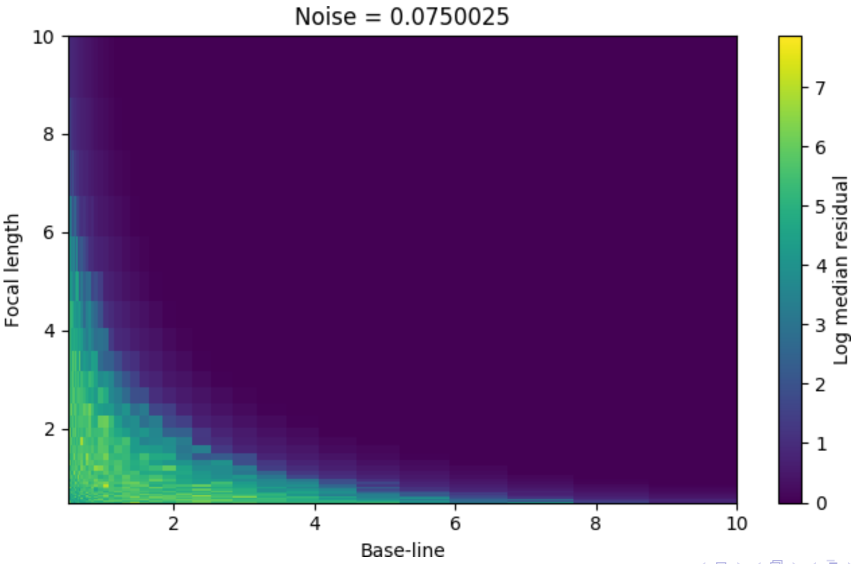
Task 2: Varying the camera base-line



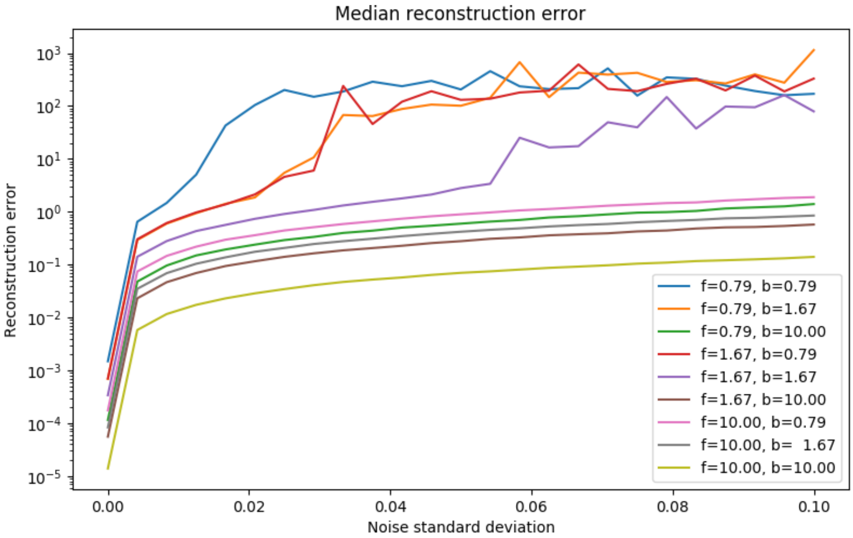
Task 2: Varying focal length



Task 2: Alternative visualisation: focal v baseline v error



Task 2: Configurations v noise

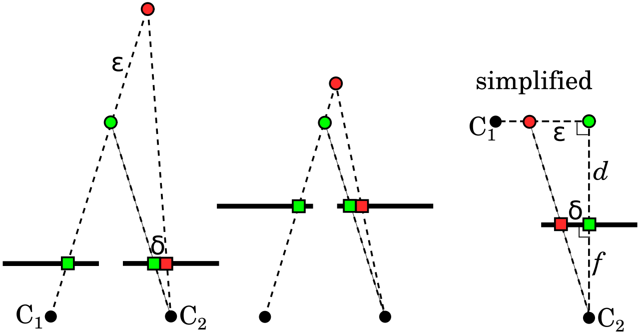


Task 2 analysis

- Your experiments should note that *in general*, error was
  1. proportional to noise
  2. *exponentially inversely* proportional to focal length and base-line
- reconstruction sensitive to small base-lines / wide field-of-view / relatively high error
- ***why do you see these characteristics?***
  - mathematical analysis wasn't required
  - 2D illustrations with discussion would suffice!

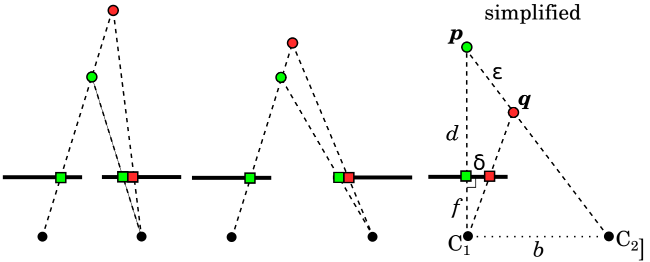
Increasing focal length

- Images scale increases in proportion to focal length
  - Because the noise is *not* scaled, the subtended angle therefore *decreases*
  - Smaller angle → smaller opposite length → smaller reconstruction error.
- error is non-linear and inversely proportional to f: $\frac{f}{d} = \frac{\delta}{\epsilon} \therefore \epsilon = \frac{\delta d}{f}$



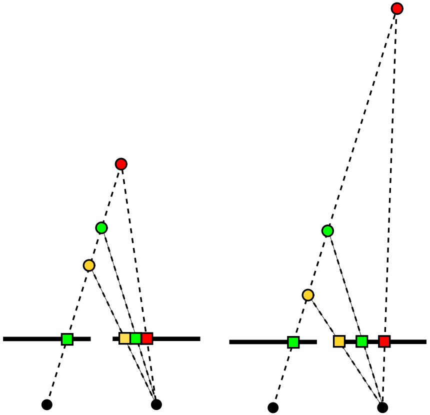
Increasing baseline

- increasing the base-line tacitly increases the angle w.r.t. the 2 cameras
- *one* approach for modelling this behaviour:
  - $\epsilon = \|\mathbf{p} - \mathbf{q}\|_2$, where $\mathbf{q} = (\mathbf{c}_2 \times \mathbf{p}) \times (\mathbf{c}_1 \times \overrightarrow{\delta})$
  - ... $\epsilon \propto \sqrt{\frac{b^2+1}{b^2+2b+1}}$
  - hence error denominator increases faster than numerator w.r.t. change in baseline



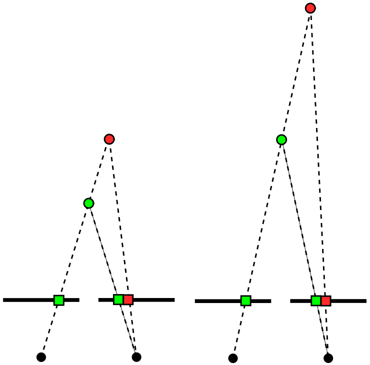
Increasing noise

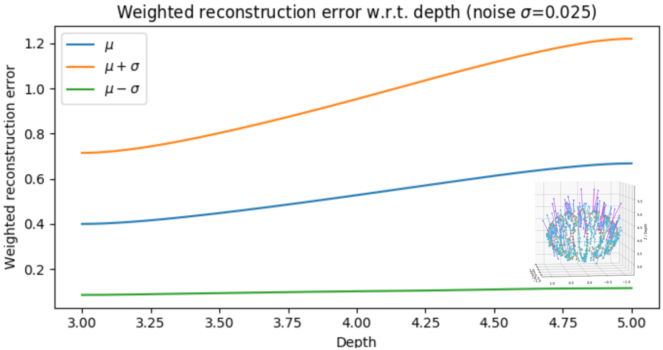
- increasing noise increases reconstruction error
- not just magnitude: the noise *direction* w.r.t. epipole affects error
  - noise *orthogonal* to the triangulation plane is less sensitive than noise *within* it
  - bonus: experiments that illustrate this by adding noise in particular directions



Function of depth

- reconstruction error is also affected by depth
- can be demonstrated empirically





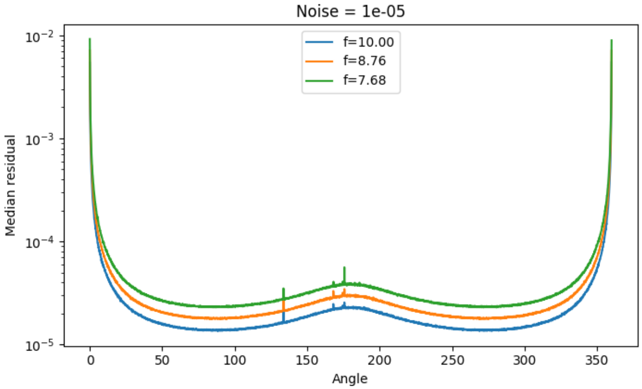
Task 3 set-up

- there are a couple of different ways of parameterising the stereo cameras:
  - camera 1 with the Identity extrinsics matrix and using trigonometry to compute the location of the second camera; or
  - simpler: put the scene at the origin and translate and rotate both cameras: $(\mathbf{R}_y \mathbf{T}_{-z})^{-1} \equiv \mathbf{T}_z \mathbf{R}_{-y}$ for fixed $-z$ for both cameras[2]
- method 1 was suggested in the assignment because it segues into Task 4 :-)
  - kudos if you recognised the second form!

[2] but with $y = 0 \rightarrow \mathbf{R}_0 = \mathbf{I}$ for the first camera!

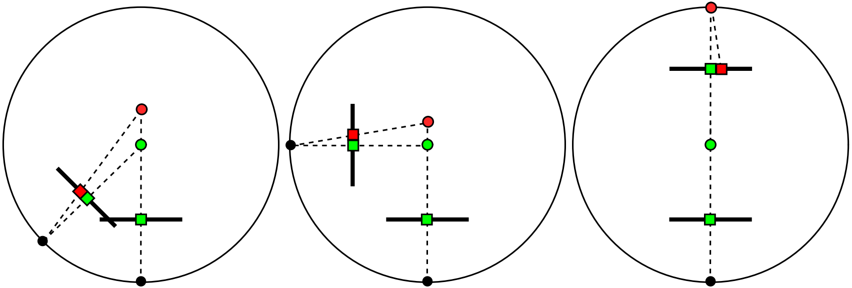
Task 3

- degenerate stereo configuration as angle $\rightarrow 0$, $\therefore$ increasing error
- minimum error when cameras are orthogonal
- error increases when cameras face each other



Task 3

- noise sensitivity is related to angle between the two cameras
  - adding noise to nearly parallel lines significantly changes where they intersect!
- facing cameras have increased risk of intersection behind one camera
  - but their images *are* different and $\therefore$ not analogous to zero baseline



Task 4

- Tasks 2 and 3 assume that the camera geometry is known exactly
  - in practice, the extrinsics (and often also the intrinsics!) are unknown, and must be estimated from the images (possibly also from EXIF meta data)
- repeat task 3, but with the added step of recovering pose from the Essential matrix
  - the problem is that the scene is recovered up to an *unknown scale*
  - could recover scale using RANSAC, or simply the median estimated scale