Date: 05/18/2021

CS 6643 - Computer Vision, Spring 2021 **Prof.James Fishbaugh**

FINAL EXAMINATION - COMPUTER VISION (CS-GY 6643)

Pruthviraj R Patil, Computer Science, MS.
New York University, Tandon School of Engineering

Email: prp7650@nyu.edu

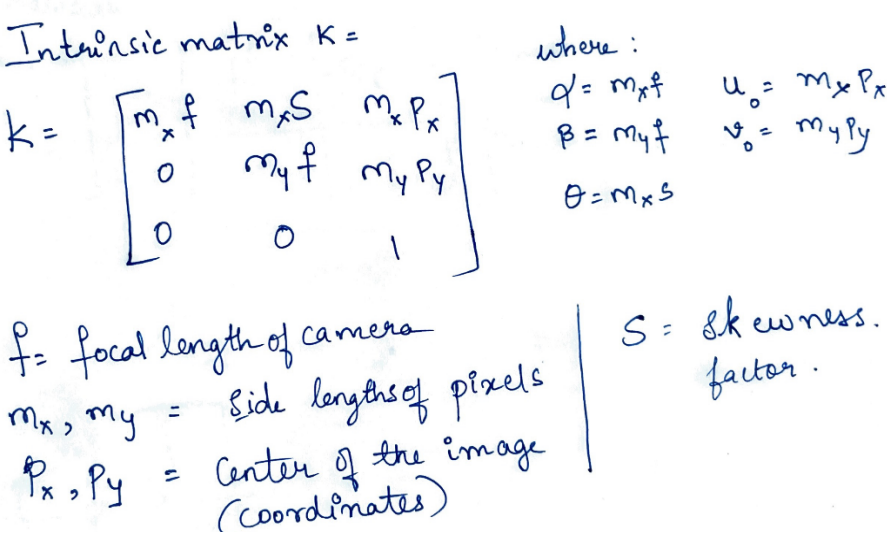
1) Camera Calibration (15 minutes)

a) At a high level, explain what is meant by camera calibration in computer vision.

Solution: Camera calibration is an application in computer vision in which the image is related with the real world. This is done by mapping the homogenous camera coordinates (2D) with the homogenous world coordinates (3D) by using some multiplying factor or in other words the Camera Matrix. This will in turn help in distortion reduction (as shown in an example of a pool table example in the class) and relate the objects in the image with the real world.

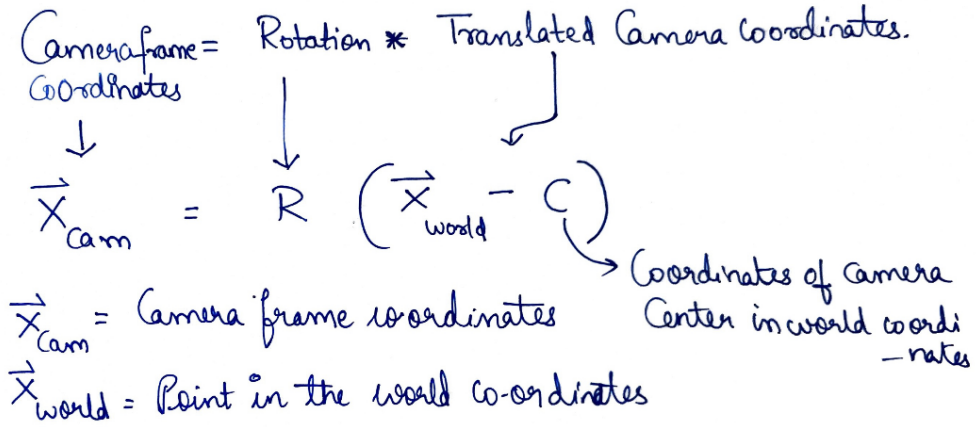
b) Explain the concept of intrinsic parameters.

Solution: The intrinsic parameters are the components of an intrinsic matrix of size (3*4) that relates the principal point (center of the image), pixel size, and focal length of the camera with each other. These are the intrinsic properties of the camera and help in mapping the digital points with the world coordinates. This matrix can be used as a camera matrix directly when it is given that the camera and the world share the same coordinate system. Eg: In the pinhole camera model. The intrinsic matrix is derived by using the concept of similarity of triangles by using the base of the pinhole camera model.

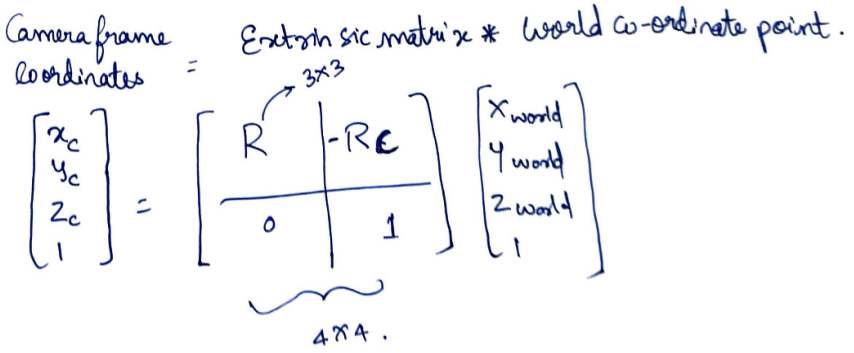


c) Explain the concept of extrinsic parameters.

Solution: In the real time**,** the camera coordinate system is not aligned to the image points. At this point, we need to align the world coordinates with the camera coordinates by rotation and translation inorder to map the camera coordinates with the world coordinates tangibly.

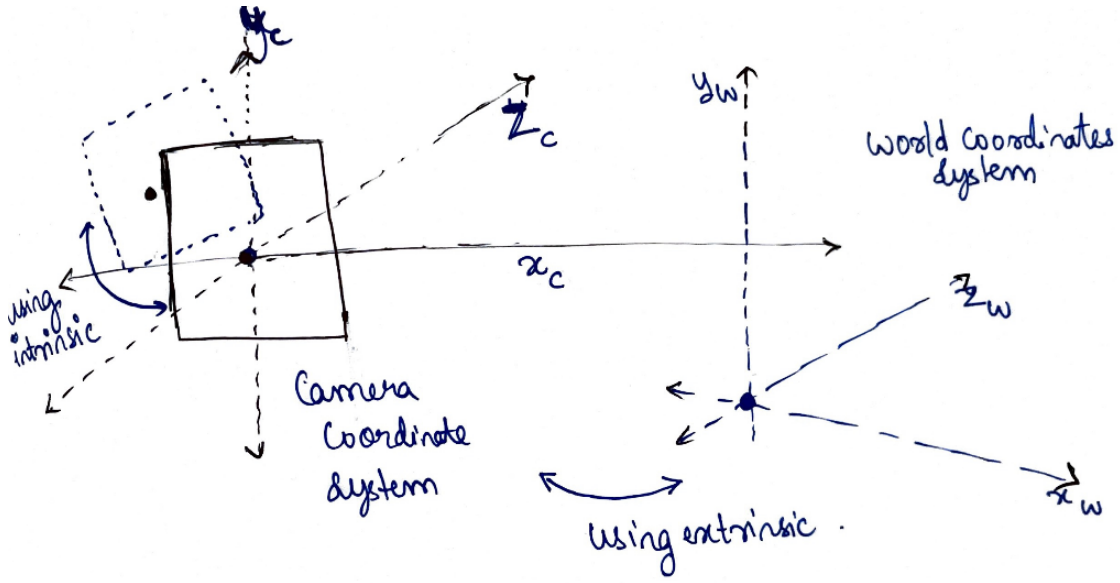


This translation and the rotation transformations combined gives us the extrinsic matrix that is of the size (4*4). The alignment further if the world coordinates are not aligned with them, then, an extrinsic matrix is used to correct those inorder to avoid the distortion. This brings in the world coordinates into the picture by knowing the center coordinates of the camera and aligning them with the center of the image with respect to the world.

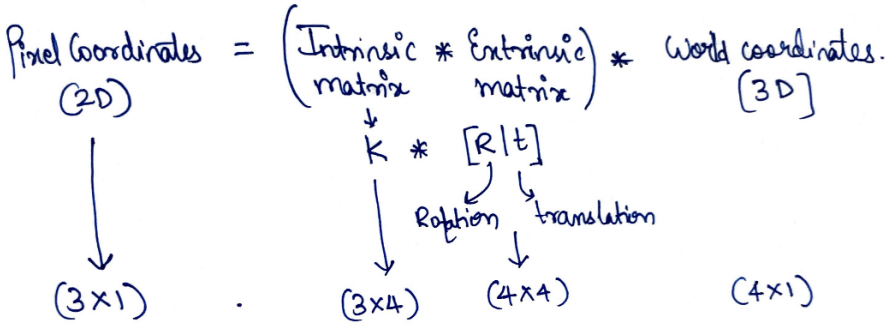


d) Describe a procedure for camera calibration (the general procedure, not the code to extract parameters).

Solution:



The camera calibration system can be done by using the intrinsic matrix and the extrinsic matrix as shown above. The intrinsic matrix checks up with the camera characteristics like focal length, pixel sizes, image center etc and adjust it according to the digital coordinates whereas the extrinsic matrix aligns the world coordinates with the camera coordinates. Thus, these can be considered as the set of internal and external transformations. Hence, multiplying them will give us the camera matrix as mentioned earlier to map the image coordinates with the world coordinates in a tangible manner.



e) Explain why camera calibration was used by youtuber “Stuff Made Here” for his automatic pool stick project.

Solution:

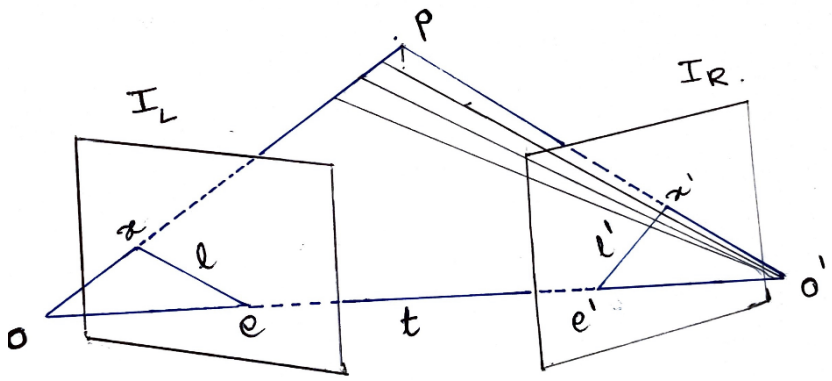
Connection between the pool ball in the image using the index of it’s center with the real world i.e over the pool table. The user plugged in the pool balls center indices into the equation consisting of the camera calibration matrix (intrinsic and extrinsic matrix) to get the coordinates

of where the ball is present in the world coordinates that he defined. This was to map the exact world coordinates on the table in order to strike the ball at the particular position using the projections given from top inorder to score accurately. Thus, reduction of distortion in mapping of the image points with the world coordinates is corrected using the camera calibration matrix in the video

2) Epipolar Geometry (25 minutes)

Solution A:

The epipolar geometry is the domain where the correspondences are found with respect to the two images that are captured using separate stereo cameras on the same epipolar plane. This correspondence can be shown by searching one image's pixel points on the corresponding epipolar line on the other image. Or to align both the camera co-ordinates we have to rotate and translate the left image onto the right.



Il and Ir are the image planes with respect to the two different stereo cameras whereas e, e' are the epipoles where the line OO' intersect these image planes. P = (x, y, z) is a real world coordinate point. The line xe=l and x'e'=l' are the epipolar lines where the points along the ray PO and PO' project on the image planes. The points e and e' are called the epipolar points and the plane POO'P is the epipolar plane that holds the two image planes and the real world coordinate P.

Solution B:

The epipolar constraints given:

$p^T E p' = 0$ where $E = [T_x] R$. → (1)

$[T_x]$: 3×3 skew sym matrix of translation; R: Rotation.

but since Rotation is zero, we consider $R = I$

∴ Eqn (1) becomes:

$$E = [T_x] \rightarrow (2)$$

but its also given that $T_x = [t_x, t_y, t_z]^T$

∴ $E = [0, 0, t_z]^T$ → (3) | as translation is only towards Z direction

Solution C:

Given point $P = (x, y, 1)^T$ & E as calculated $E = [0, 0, t_z]^T$

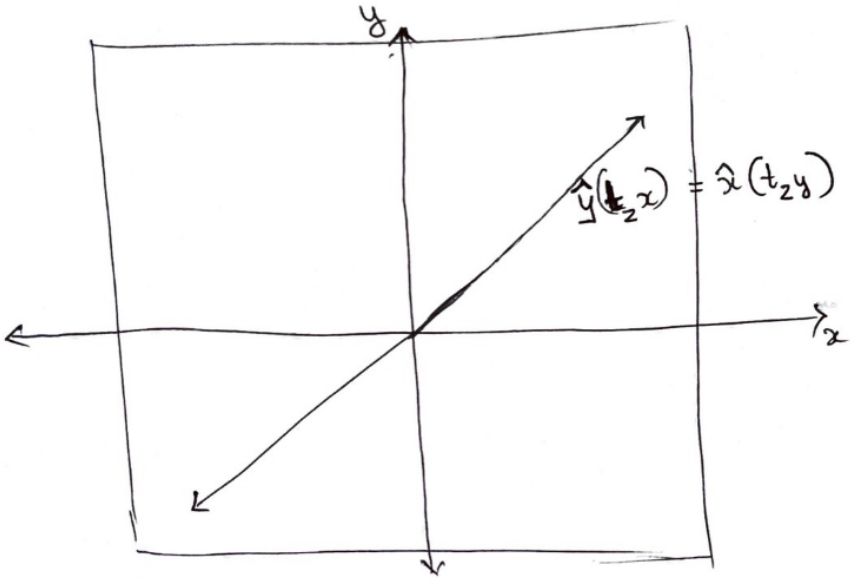
$$\begin{bmatrix} 0 \\ 0 \\ t_z \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & -t_z & 0 \\ t_z & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\therefore \Rightarrow \begin{bmatrix} -t_z y \\ t_z x \\ 0 \end{bmatrix} = l_R = \text{epipolar line.}$$

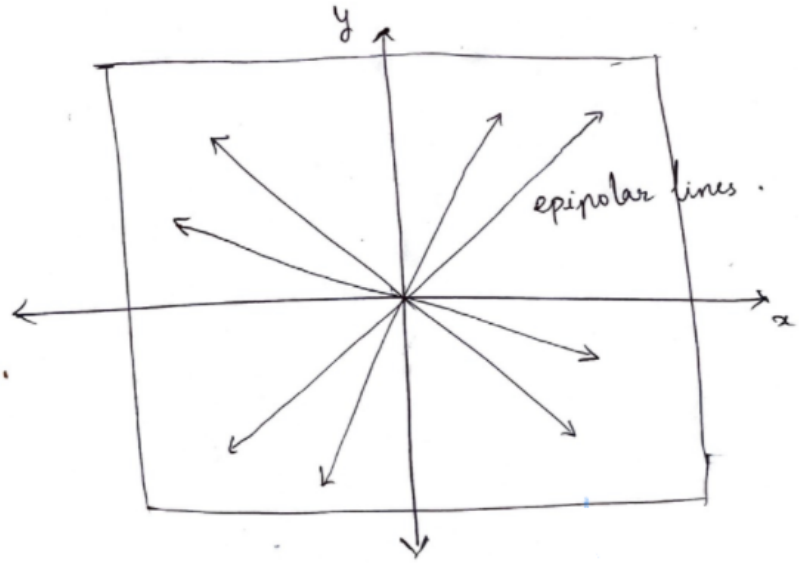
∴ this line is represented as $a\hat{x} + b\hat{y} + c = 0$

$\Rightarrow \hat{x}(-t_z y) + (t_z x)\hat{y} = 0$

$\therefore \boxed{(t_z x)\hat{y} \; (t_z y)\hat{x}}$



Solution D



When the camera moves in the z direction, the elements in the image get enlarged or get reduced. The pixel between the images can move either move towards the center or away from the center as there is no translation but there is only rotation. Thus the result is expected as above. The epipolar line is the one that connects the points towards the center, it would be the set of locations where we can search for the points for the same image patch

3) Stereo Vision (15 minutes)

Solution A:

Disparity is the distance between one point on a stereo image and its corresponding point on another stereo image on the same epipolar plane. So, as the depth increases the disparity decreases. This is because the object which is significantly far away from two stereo cameras tends to occur at the same position in its images captured by both the cameras because the disparity tends to be insignificant. This can be given by the equation:

$$Depth = \frac{focal\ length\ of\ the\ cameras * distance\ between\ them(centers)}{disparity}$$

No, the depth is not the same as disparity. Moreover, they are inverse to each other as shown in the above equation.

Solution B:

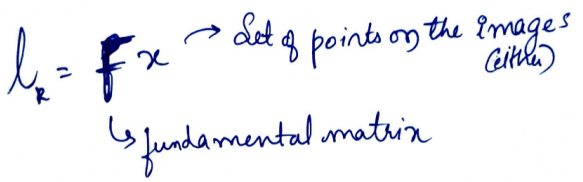
Steps to compute the dense disparity map:

1. Find the fundamental matrix by using the points on the right and the initial image (left). This can be shown in the equation below. There will be 9 unknowns in the Fundamental matrix F. So, 9 points at least are needed to solve these points.

$$x'_m F x_m = 0$$

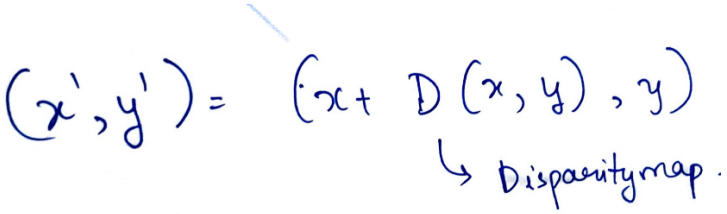
where x'_m = points on Righting
x_m = points on left img.
F = fundamental matrix.

2. So, after that, we get the epipolar lines of the equation given below:



3. Getting these, if we translate the left image to match the right image in order to achieve the continuous epipolar lines.

4. It can be intuited that if we subtract the images along x direction, we can get the disparity map as we can get exactly how much the image has translated in the x direction.



4) Optical Flow (25 minutes)

Solution A:

given :

Brightness constancy :

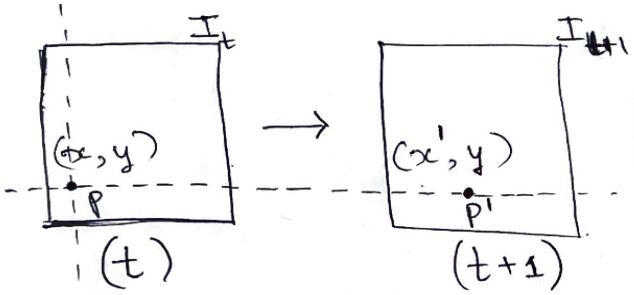
$$I_x u + I_y v + I_t = 0$$

$$\frac{dI}{dx} u + \frac{dI}{dy} v + \frac{dI}{dt} = 0$$

Soln:

In the case of 1-dimension, we assume :

$$I(x,t) = I(x+u\delta t, t+\delta t) \longrightarrow (1)$$



here, $\frac{dy}{dt} = 0$

I_t = Image at time t,

I_{t+1} = Image at time $t+\delta t$

Note : "y" is unchanged as the change is only in "x" direction

∴ the point given at I_t i.e. $P = (x, y)$ shifts to P' in the image $I_{t+\delta t}$ to $P = (x+\delta x, y)$

$u\,dt$ = displacements vector component

u = optical flow component → [speed / velocity]

from Eqn. ①,

$$I(x,t) = I(x+u\,dt, t+dt)$$

According to Taylor series,

$$f(x+dx, y+dy, t+dt) = f(x,y,t) + f_x(x,y,t)dx + f_y(x,y,t)dy + f_t(x,y,t)dt$$

(partial derivatives)

∴ we can rewrite brightness constancy as:

$$I(x,t) + I_x(x,t)dx + I_t(x,t)dt = I(x,t)$$

$$\therefore \quad I_x(x,t)dx + I_t(x,t)dt = 0$$

dividing by dt, we get

$$I_x(x,t)\frac{dx}{dt} + I_t(x,t) = 0$$

↳ velocity "u" as mentioned earlier.
is a optical flow component and
y-velocity is absent as it is in "1D" (flow)

$$\therefore \quad I_x \frac{dx}{dt} + I_t = 0$$

$$\therefore \quad \boxed{I_x u + I_t = 0}$$

is the brightness Constancy equation

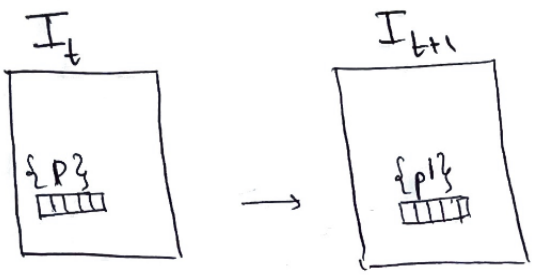
Solution B:

b. we know that $I_t = \frac{dI}{dt}$ and as mentioned earlier $u = \frac{dx}{dt}$ & in this case, $\frac{dy}{dt} = 0$

as the brightness constancy equation is $I_x u + I_t = 0$,
the solution lie on a line $ax + b = 0$

∴ we can solve for u using Lucas method by assuming a flow within a patch or neighborhood. [say 5×1]
So, the neighboring pixels have same displacement under 5×1 pixels.

Considering 5×1 image patch, we can see that:



$$\therefore \quad I_x(p_1)\,u = -I_t(p_1)$$
$$I_x(p_2)\,u = -I_t(p_2)$$
$$I_x(p_3)\,u = -I_t(p_3)$$
$$I_x(p_4)\,u = -I_t(p_4)$$
$$I_x(p_5)\,u = -I_t(p_5)$$

$$\therefore \quad \begin{bmatrix} I_x(p_1)\,u \\ \vdots \\ I_x(p_5)\,u \end{bmatrix} \cdot [u] = - \begin{bmatrix} I_t(p_1) \\ \vdots \\ I_t(p_5) \end{bmatrix}$$

$A_{(5\times 1)}$; Unknown $= \vec{U}$ (1×1) ; $b_{(5\times 1)}$

$\therefore$ A & B are known already

$\therefore$ we can solve u by using least sqr method.

where $\boxed{\vec{U} = (A^T A)^{-1} A^T b}$

Horn Schunck Approach to find U:

* (1) Due to Aperture problem, we need to use Horn-Schunck process to calculate optical flow.

So, instead of considering neighborhood points that denotes "constant" flow, smoothness of flow is considered. This is because the real objects are rigid or elastic move together coherently.

So, brightness constancy and the smoothness on the velocity field are enforced. Hence every time we calculate them, the cost function governed by those factors has to be minimized.

where

$$u^{k+1} = \bar{u}^{k} - \frac{I_x (I_x \bar{u}^{k} + I_t)}{\lambda^2 + I_x^2}$$

u^{k+1}: next epoch's "u" value

$\bar{u}^{k}$: previous epoch's "u" $_{avg}$ value

where $u_{avg} = \frac{1}{4}\left[u_{(i+1,j)}, u_{(i-1,j)}, u_{(i,j+1)}, u_{(i,j-1)}\right]$

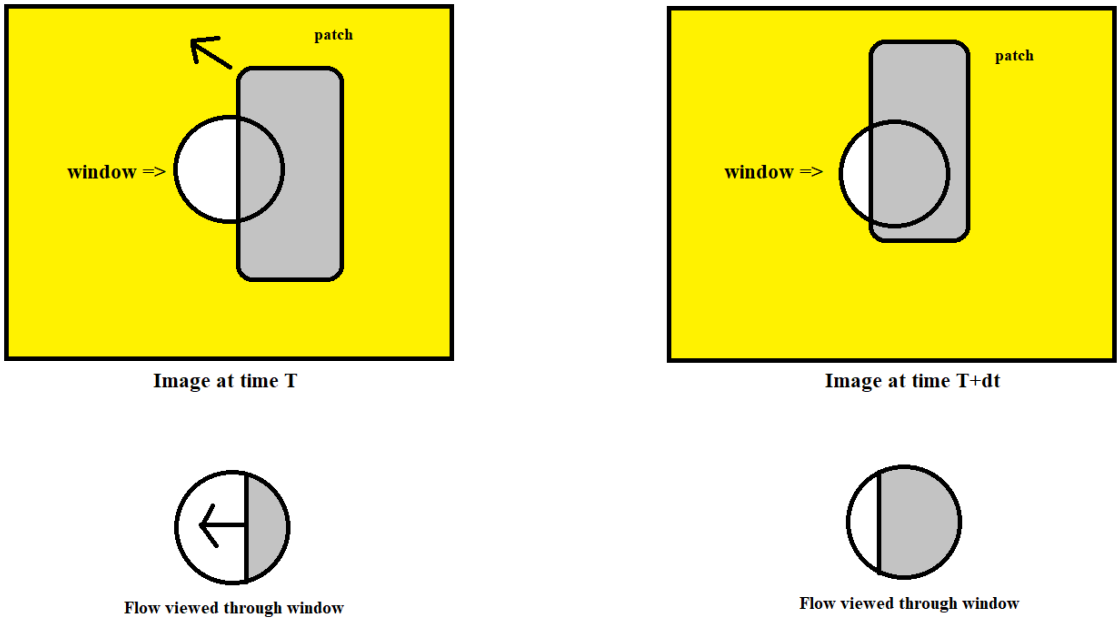
u_{avg}: local avg

This is done for either given number of epoches or till the cost function converges. when minimized in every epoch.

* Cost function $= \min_{u} \sum_{(i,j)} \{ \underbrace{E_s(i,j)}_{\text{smoothness}} + \lambda \underbrace{E_d(i,j)}_{\text{Brightness constancy}} \}$

Solution C.

The aperture problem is a limitation of Lucas method to calculate the optical flow u. This is noted by viewing the optical flow of the image patch through a small aperture (window) in an image as shown below. Even though the image patch is moving with respect to both x and y direction as shown on the LHS image, the change as seen through the window in the RHS is ambiguous as it shows that the change occurred only in the x direction.So, the dependency of the neighborhood window patch size is the major limitation for the Lucas method inorder to determine the actual flow.



(Image prepared using MS Paint)

Solution D.

There are multitudinous limitations of the Lucas-Kanade method to calculate the optical flow.

- The frame (neighborhood) size of the do impact in a very decisive manner because of many factors. In the case of 2-D optical flow, If we use 5*5 neighborhood window, we get total of 25*3 equations to calculate u, v. Hence, more the frame size (say size = 11), the magnitude of the optical flow between the two consecutive temporal images is seen more. This results in the vectors to be seen in a proper direction. So, it is difficult to speculate the proper number of neighborhoods to be considered in order to gain viable results

- In the case of the lower sized window, the magnitude of the optical flow will not be not properly visible as it may result in the aperture problem. Hence, the vectors can point in different directions.
- However, if the neighborhood window size increases more, then there can be too much of the generalization given to the direction of the movement hence, we can miss the minute details of temporal change but only get the bigger picture of the flow and also the time complexity of the process too increases a lot as the number of equations to calculate the optical flow components also increase accordingly.

The advantages of the Lucas process are:

- Faster execution is the major advantage of the process because ulike Horn Schunk process, this does not require Energy minimization process if the neighborhood size is considerable enough.
- This process involves the neighborhood size as a parameter, this can calculate the optical flow in the local points unlike Horn Schunk. This doesn't care about the other neighborhoods unlike Horn Schunk that cares about the overall image in order to minimize the Energy function.

According to me, the major disadvantage of this process is the aperture problem. This is a major caveat where even though the image patch is moving with respect to both x and y direction, the change as seen through the aperture window is ambiguous as it shows that the change occurred only in either x or y direction.

And the major advantage of the Lucas method is fast execution. Even though it is hard to speculate about the neighborhood size parameter, if it is considered in a viable manner, then we can gain viable results in a lesser amount of time.

Solution E.

There are multitudinous limitations of the Horn-Schunck method to calculate the optical flow.

- The value of λ does play the major role in the calculation of optical flow vectors. The values of u and v in the process are calculated by subtracting their values in previous epochs by the factor which is inversely proportional to λ. This makes the process converge more early if the value of lambda is decreased. Because the new values of U and V are used to calculate the Energy value which is a combination of the Smoothness factor of optical flow and the Brightness constancy factor (that acts as a cost function).
- However, if the lambda value is decreased too much, then too the energy value can shift to the other side, becoming more negative hence yielding the incorrect results. Also, if the lambda value increases a lot, the number of epochs needed might be even more, this

might also cause problems of increase in time complexity. This process would be similar to Gradient descent.

On the positive side, Horn Schunk process has these advantages:

- Minute moments can be detected. This is a global optimization process that requires several epochs in order to converge the cost function to get the optimized value. Also, in the process the average optical flow parameters are calculated by taking the neighbors of each and every data point. Hence, this method can check the minute details that occur in the temporal images.
- Overcomes the aperture problem by using the Smoothness factor in the energy function. It has the dense points where lots of information can be stored. This can be improvised by using the process of minimizing the energy function.

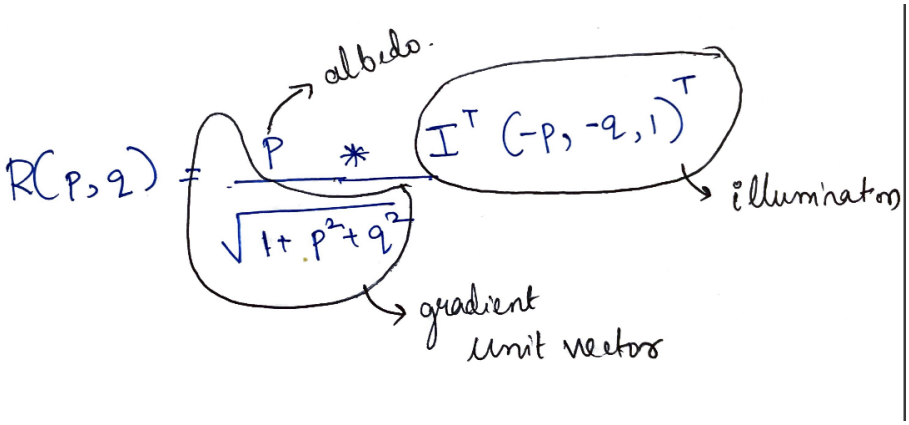
According to me the time complexity of the process is the biggest caveat because this completely depends on lambda value. This is a global optimization process similar to the Gradient descent algorithm. Thus, if the local minimum is located as the final result then the optimized result may not be acquired.

The capacity to find the minute changes and to fix the aperture problem in the temporal difference of the image overall is the biggest advantage of this process. Because, if this isn't the problem, then the Lucas method is preferred over this.

5. Shape from Shading (25 minutes):

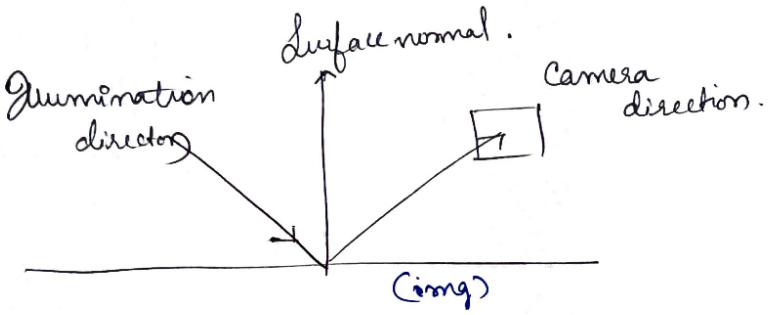
Solution A:

Shape from shading is a process in which the shading of the object is used in order to determine its 3-D shape. The characteristics of the shading namely the Surface geometry, Reflection of the object (Albedo), Illumination over the object (position, direction and intensity), and finally the Camera parameters are harnessed inorder to work out the process.



From the above equation we get the shape of the image R(p, q) derived from the three of the four characteristics (camera params are neglected). After that we compare that with the original image if we are accurate to that or not. If not then we iterate the process again and adjust the p and q values taken from the surface of the image in every epoch by minimizing the cost function.

Solution B:



Lets say the surface equation is :

$$z(x, y) = z$$

partial derivatives of z w.r.t x & y will be:

$$\frac{\partial z(x,y)}{\partial x} = p \quad , \quad \frac{\partial z(x,y)}{\partial y} = q$$

The normal vector is a unit vector that stands normally (90°) with respect to the surface which can inturn be calculated by cross product of gradient with a unit vector.

$\therefore$ n = normal vector ^{T}.

$$= \frac{1}{\sqrt{p^2 + q^2 + 1}} (-p, -q, 1)$$

Solution C:

The parameters in the cost function and the components of it are mentioned in the equation given below:

The cost function given:

$$E = \iint [I(x,y) - R(p,q)]^2 dx dy + \lambda \iint (p_x^2 + p_y^2 + q_x^2 + q_y^2) dx dy$$

$I(x,y)$ = observed image

$R(p,q)$ = Calculated model.

λ = weight

$p_x^2, p_y^2, q_x^2, q_y^2$ ⇒ second derivatives.

The first part of Energy function deals with the data matching whereas the second part deals with smoothness of image observed.

The data matching component deals with how the calculated image is close with the original image with respect to each and every gradient p, q. So, Summation of this over all the image pixel gradients gives us the data matching energy function. The latter part of the Energy function deals with the smoothness of the observed image. This can be calculated by summing all the squares of the second derivatives. This is known to be the Smoothness energy.

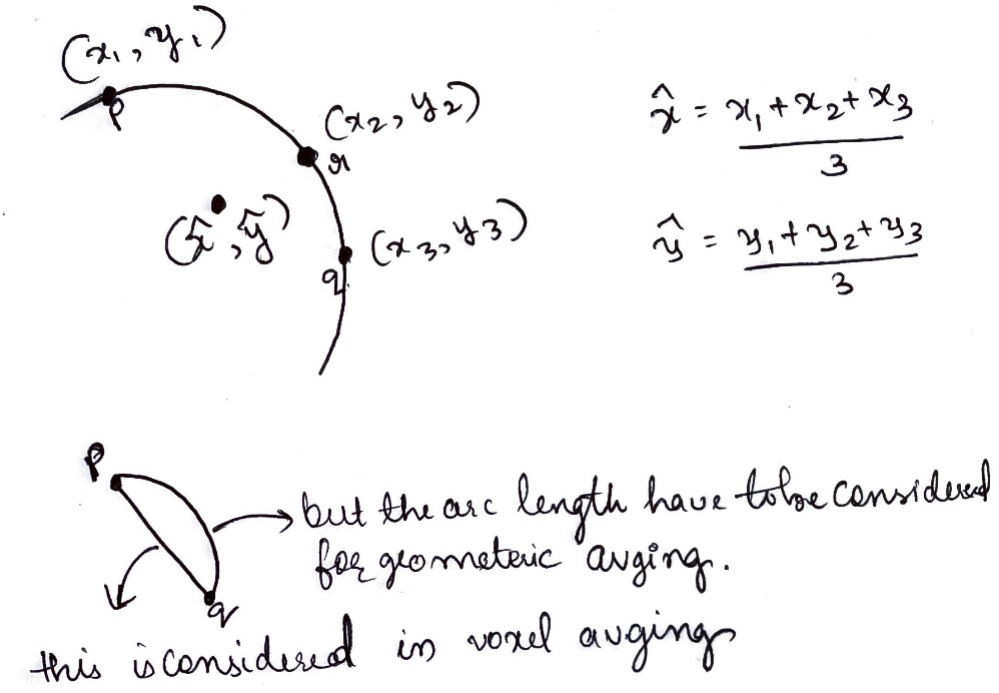
Further, λ is called as a ratio factor. This is used to decide what weightage to give to the smoothness over the data matching aspect to minimize the cost function. Thus, if weightage to the smoothness to be high then λ is valued higher else lower. As we minimize the cost function after each iteration. Then we can partially derive E with respect to p and q for every iteration. The values of p and q are used to determine the surface gradient of the observed image. As mentioned earlier and also through the earlier figure, we can see that the surface gradients give out the normal and height of each point in the image domain that are in turn used to 3-D model the surface of the observed image.

6. Alternative View of Average (10 minutes):

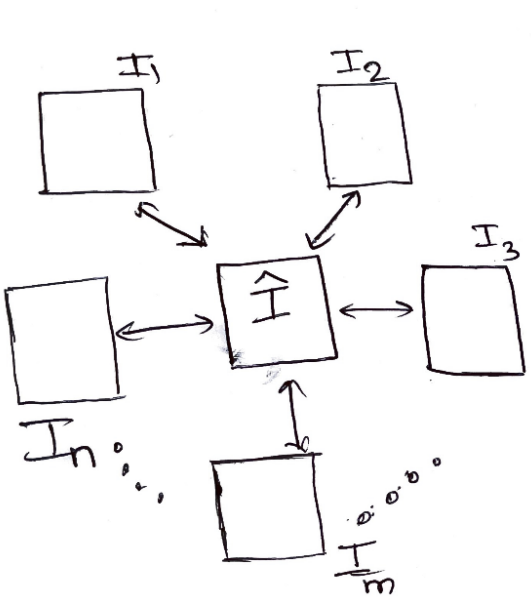
Solution A:

Alternative view of average is the special case of the voxel average finding process (Euclidian). It is nothing but the generalization of the usual process of calculating the averages. The geometric average is nothing but the calculation of the Frechet mean of the Images. The major difference between the Euclidean average method and the Frechet mean method is that the latter method is a process to get an average based on the shape of the object but not the image itself.

For example, considering the points on the arc of the circle, the intuitive average between these points must lie on the circle itself. But, in the case of Euclidean average gives the average far from them. Hence, length of the arc must be considered in this case to calculate the average but not the actual distance between the data points.



The geometric averaging is considered to be the unbiased Atlas building process. The figure given below suggests the images whose average is calculated in the center.



Idea: to minimize the sum of deformations from the observations to atlas. or vice versa.

done by: using diffeomorphism

This can be done by using the equation:

$$E(\phi_i, \hat{I}) = \sum_{i=1}^{N} \| \phi_i(\hat{I}) - I_i \|^2 + \lambda \, Reg(\phi)$$

$\sum_{i=1}^{N}$: sum over observations. $\| \phi_i(\hat{I}) - I_i \|^2$: image similarity (Normalized cross correlation). I_i: Ith observation. $\lambda \, Reg(\phi)$: Regularity

Here, ϕ(I) is known as the diffeomorphism which encodes thee variability i.e. where in calculating the average of the images, two images can be linked by passing them through the atlas as for example : $\phi_2^{-1}(\phi_1(I_1))$ in which I_1 is bought into I_2 space.

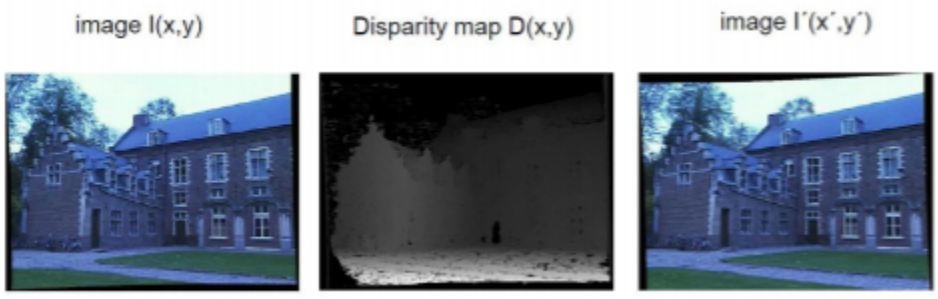
Solution B:

The scientist who pioneered the concept of studying shape variability through the mathematics of transformations is D'Arcy Wentworth Thompson.

7) Miscellaneous (5 minutes)

Solution A:

According to me, the broadly useful topic must have a wide horizon/scope in it that branches out to multiple domains under it to explore. Such a topic that is discussed in the course is Disparity and Depth Mapping where the depth parameter in the stereo images is taken into consideration to check the relation between them.



(Image taken from slides)

For instance, in the case of the image classification process where the number of classes are more in number and the image classes almost resemble each other in 2D when viewed in that perception, there will be a requirement of a set of differently orientated stereo images in 3D including the depth as a parameter. This will not only help to classify the ambiguous classes in a tangible manner, but also help in increasing the amount of data required to train the model inturn increasing the robustness of the classification model.

This is why the disparity and depth mapping is one of the broad topics that is discussed in the course.

STUDENT HONOR PLEDGE

I pledge on my honor that I have not given or received any unauthorized assistance on this exam.

(Pruthviraj R Patil, N16324281)