CS 280 Computer Vision
Midterm Exam
Thursday, April 7th, 2016

Trevor Darrell, Alexei A. Efros
Richard Zhang, Deepak Pathak, Jeff Donahue

First	
Last	
SID	

Name of neighbor to your:

Left	
Right	

For staff use only:

Q1. True/False	**__/14**
Q2. Short answers	**__/15**
Q3. CNN parameters	**__/18**
Q4. Binocular stereopsis	**__/10**
Q5. Comparing heights	**__/12**
Q6. Image processing	**__/12**
Q7. Pyramids	**__/5**
Total	**__/86**

Q1 [14 pts] True/False

For each of the following, indicate whether the statement is TRUE or FALSE.
Each True/False question is worth 1 point.

T **F** Parallel lines in an image will remain parallel after applying a homography

T **F** Perspective projection from a 3D point in the world to a 2D point in the image is linear in Cartesian coordinates

T **F** Vanishing points of all lines on a plane lie on a line.

T **F** In a Camera Obscura, the smaller the diameter of the pinhole, the sharper the image becomes.

T **F** Rotation, Affine and Projective transforms in Euclidean space can be expressed simply as a linear map in homogeneous coordinates.

T **F** Filters approximating the oriented cells in V1 discovered by Hubel and Wiesel emerge naturally if you learn a sparse code on a dataset of natural images

T **F** Filters approximating the oriented cells in V1 discovered by Hubel and Wiesel emerge naturally in the first layer of a CNN trained for ImageNet classification

T **F** Given only a brightness histogram [256 bins] of an image, one can estimate an orientation histogram [6 orientation bins] for that image.

T **F** A CNN implements a linear function

T **F** Optimizing parameters for a CNN is a non-convex problem

T **F** During a feed-forward pass, *any* convolutional layer can be implemented as a fully connected layer

T **F** During a feed-forward pass, *any* fully connected layer can be implemented as a convolutional layer

T **F** It is possible to model varying size sequences with varying spatial size inputs (i.e. each element of the sequence is of different size) using a recurrent LSTM unit.

T **F** RANSAC is used to get rid of outliers in situations when there are not enough point correspondences for least-squares estimation to work well.

Q2 [15 pts] Short Answer

(a) [2pts] Express the point where 2-D parallel lines intersect in homogeneous coordinates (assume any notation).

(x,y,0)

(b) [2pts] How many degrees of freedom are there in perspective projection transform between two images?

There are 8 degrees of freedom in a perspective projection transform between two images. Given point p = (x,y,1) and p' = (x,y,1) in a second image, we would like to solve for P' = H*p, where p' is P' normalized over the last coordinate. Matrix H has 9 values, but one must is fixed to prevent arbitrary scaling.

(c) [2pts] $\mathbf{E}\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \mathbf{M}\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$ What is matrix $\mathbf{M}$ for orthographic projection?

M =
[1 0 0 0]
[0 1 0 0]
[0 0 0 1]

(d) [2pts] Derive the rank of the Essential Matrix $\mathbf{E}$.

E=[t]R --- Here t is of rank 2 (because the direction along t forms a rank 1 null space of t; and other two orthogonal directions constitute rank 2) and R is full rank. Thus E is rank 2.

(e) [2pts] A projective transformation between two images $\mathbf{X}$ and $\mathbf{X}'$ can be modeled by transforming each point $\mathbf{p}$ in one image to a point $\mathbf{p}'$ in a new image using $\mathbf{p}' = \mathbf{H}\mathbf{p}$, where $\mathbf{H}$ is the homography. Briefly explain, why we can't do the same thing with the Fundamental Matrix $\mathbf{F}$, to help transform each point $\mathbf{p}$ in the Left image to a point $\mathbf{p}'$ in the Right image: $\mathbf{p}' = \mathbf{F}\mathbf{p}$?

There is an ambiguity in this case, since a point in one image maps to the *epipolar line* in the other image.

(g) [3pts] What is the typical loss function used to train Siamese Networks? Define each term in the expression and the importance of each component.

L(x_1, x_2, y) = y * max(||f(x_1)-f(x_2)|| - m1,0) + (1-y) * max(m2 - ||f(x_1)-f(x_2)||,0)

y is whether both examples have the same loss
f is the network
m1 is the margin if they have the same label
m2 is the margin if they have different labels
if they have the same label, we want them to be closer together
if they have different labels, we want them to be further apart

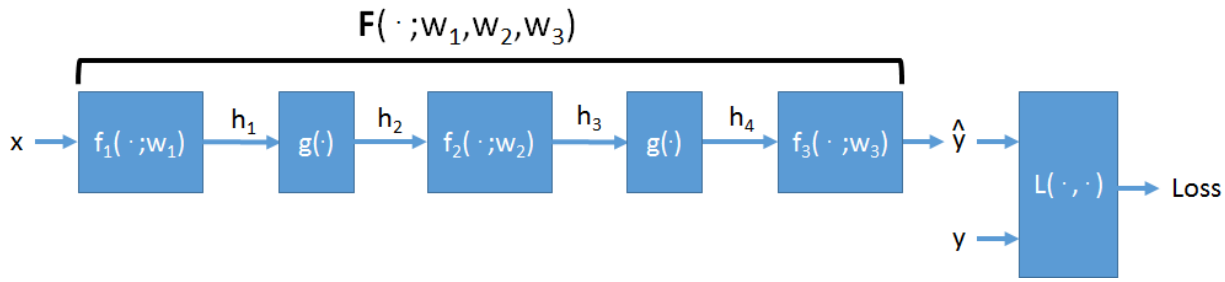
(h) [1 pt] I would like to take a derivative of discrete signal $x[t]$. What convolutional kernel can I use?

[1 -1] or [1 0 -1] or flipped

(i) [1 pt] I would like to take the second derivative of discrete signal $x[t]$. What convolutional kernel can I use?

[1 -2 1] or whatever is above convolved with itself

Q3 [18 pts] CNN Computations



Given input x and ground truth y we have a network F with parameters w_1, w_2, w_3, drawn above. During training, we penalize the network with loss function $L(F(x), y)$.

In terms of individual layer Jacobians (e.g., $\frac{\partial h_3}{\partial h_2}, \frac{\partial h_2}{\partial h_1}, \frac{\partial h_3}{\partial w_2}$), write formulas for the following:

(a) [3pts] Write $\frac{\partial L}{\partial x}$

dL/dx = dL/dy_hat * dy_hat/dh4 * dh4/dh3 * dh3/dh2 * dh2/dh1 * dh1/dx

(b) [3pts] Write $\frac{\partial L}{\partial w_2}$

dL/dw2 = dL/dy_hat * dy_hat/dh4 * dh4/dh3 * dh3/dw2

We have an input image of spatial resolution 224x224 and channel dimension of 3. The first layer of the CNN has 64 convolutional filters of spatial size 3x3. The filters are evaluated with stride of 2. The input image is zero-padded with one (1) extra pixel around all borders (resulting in a padded image of size 226x226). (There are no groups or dilations).

(c) [3pts] What are the output dimensions for a single image after applying the first convolutional layer?

112x112x64
Spatial size is (X-Xkern+2*PAD+1)/S = (224-3+2+1)/2

(d) [3pts] How many parameters are there in the convolutional layer, including a bias term applied to the convolution output? You may leave your answer as an algebraic expression.

Xkern*Ykern*Chnout*Chnin + Chnout = 3*3*64*3 + 64

We have an input layer of spatial resolution 32x32 and channel dimension of 3. The first layer of our network is fully connected with output size 1024. The second layer of our network is a ReLU nonlinearity. The third layer of our network is a fully connected layer with output size 1024.

(e) [3 pts] How many parameters are there in the first fully connected layer, including a bias term applied to the fully connected output? You may leave your answer as an algebraic expression.

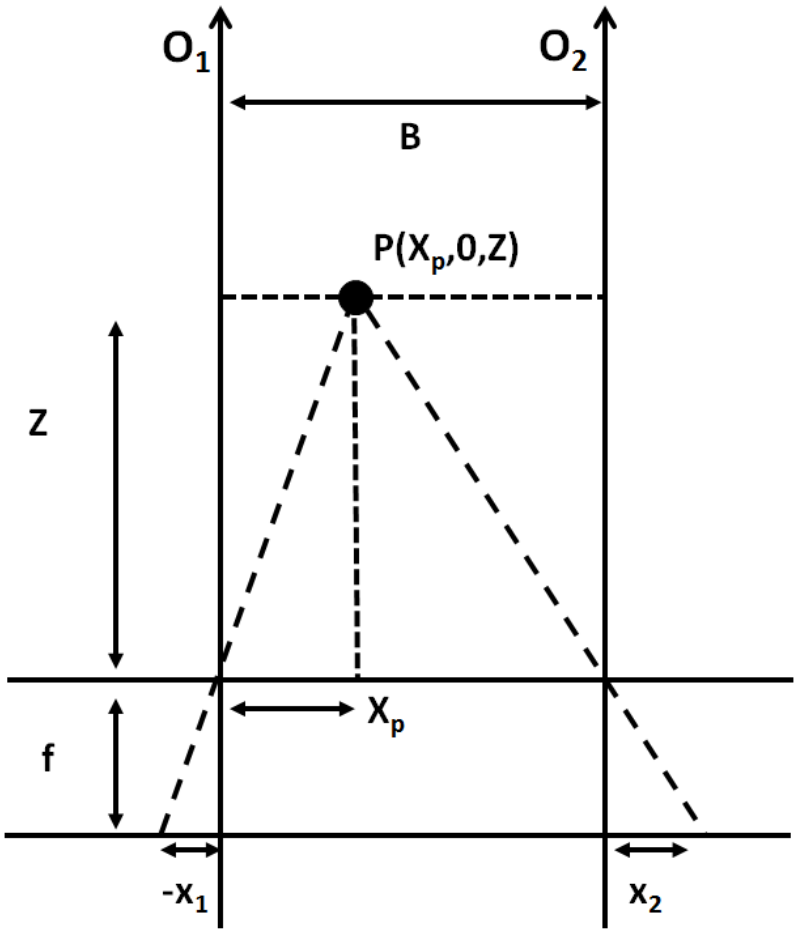
Xkern*Ykern*Chnout*Chnin + Chnout = 32*32*1024*3 + 1024

(f) [3 pts] How many parameters are there in the last fully connected layer, including a bias term applied to the fully connected output? You may leave your answer as an algebraic expression.

Xkern*Ykern*Chnout*Chnin + Chnout = 1*1*1024*1024 + 1024

Q4 [10 pts] Binocular Stereopsis

We have two optical axes $\mathbf{O}_1$, $\mathbf{O}_2$ offset by baseline distance B. Point $P(X_p, 0, Z)$ projects onto points $(x_1, 0)$ and $(x_2, 0)$ on the image planes.



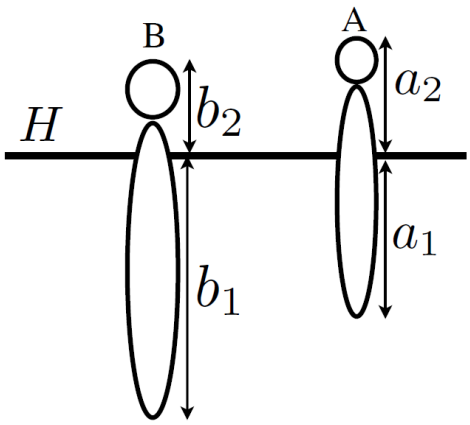
(a) [5pts] Write values x_1, x_2 in terms of the given constants.

x1 = -f * Xp / Z
x2 = f * (B-Xp) / Z

(b) [5pts] What is the estimated depth Z, given focal length f, baseline B, and difference in project points $\Delta x = x_2 - x_1$?

delta_x = x2-x1
= f * (B-Xp) / Z - f * Xp / Z
= f*B/delta_x
Z = f * B / delta_x

Q5 [12 pts] Comparing Heights



Consider the image of person A and person B standing on the ground plane, as taken by a perspective camera of focal length f. H is the horizon line ($y = 0$). a_1 is the distance between the A's toes and the horizon, and a_2 is the distance between A's head and the horizon in units of pixels. Similarly for person B. Distances a_1, a_2, b_1, b_2, are expressed in the camera plane. Suppose A's height is h feet.

(a) [3 pts] From the picture, who is taller? Briefly explain why.

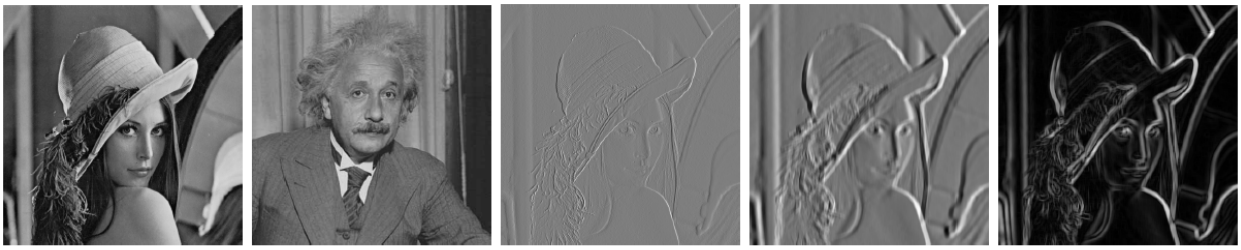
Person A is taller. Person A is standing behind person B, since you can draw a line from person B's feet to person A's feet and extend to the horizon. If they were the same height, then their feet and their heads would lie on parallel planes. In that case, you could draw a line from person B's head to person A's head to the same point on the horizon. However, if you draw a line from person B's head to person A's head, it actually goes up and doesn't even meet the horizon. Thus, person A is taller.

(b) [9 pts] Give expressions for the following in terms of f, a_1, a_2, b_1, b_2, h. Show your work briefly.

How many feet above the ground is the camera?	h_c = h * a1 / (a1+a2) The horizon line is straight ahead of the camera. The height of the camera is equal to the percentage of person A which is above the horizon times his height.
How tall is person B (in feet)?	h_b = h_c * (b1+b2)/b1 = h * a1/(a1+a2) * (b1+b2)/b1 We know portion b1 corresponding to camera height h_c. We then add on the portion of Person B's height which is above the horizon line.
Distance (along the z-axis) from the camera to person B (in feet)?	b1+b2 = f * h_b / Z Z = f/(b1+b2) * h_b = f*h/b1 * a1/(a1+a2) Height of person B projects onto b1+b2 in the image plane. Use projection equation and then solve for Z.

Q6 [12 pts] Image Processing

Consider the input images A and B. Assume that *g* is a 2-D Gaussian filter, $d_1 = (-1 \quad 1)$ and $d_2 = (-1 \quad 1)^T$.



(a) Image A (b) Image B (c) Image C (d) Image D (e) Image E

For each of the following desired outputs, write out the image processing operations that you would do with the images A and B using the filters g, d_1 and d_2 to get the desired output. The first has been done for you as an example. You can ignore edge artifacts.

Smooth image A	$A * g$
Remove high frequencies from image A	**A*g**
Remove low frequencies from image A	**A + lambda*(A-A*g)**
Hybrid image which looks like A from far off and looks like B from close by	**A*g + B-B*g**
Generate image C from image A	**A*d1**
Generate image D from image A	**A*g*d1**
Generate image E from image A	**Sqrt((A*g*d1)^2 + (A*g*d2)^2)**

Q7 [5 pts] Pyramids

We are given an image I. Let $G_1 = I$, G_2, and G_3 be the resulting images from a 3-levels Gaussian Pyramid constructed from I. Equivalently, let L_1, L_2, L_3 be the resulting images from a 3-level Laplacian Pyramid constructed from I.

Assume that the original image contained all spatial frequencies. Label the resulting images as containing H (high), M (medium), and/or L (low) frequencies. You can use none/one/many labels per image.

G_1: ___H M L____

G_2: _____ **M L**_____

G_3: _____ **L**_____

L_1: _____**H**_____

L_2: _____**M**_____

L_3: _____**L**_____