Akash Chaurasia
achaura1@jhu.edu
JHU CS461 Fall 2019

Homework 3

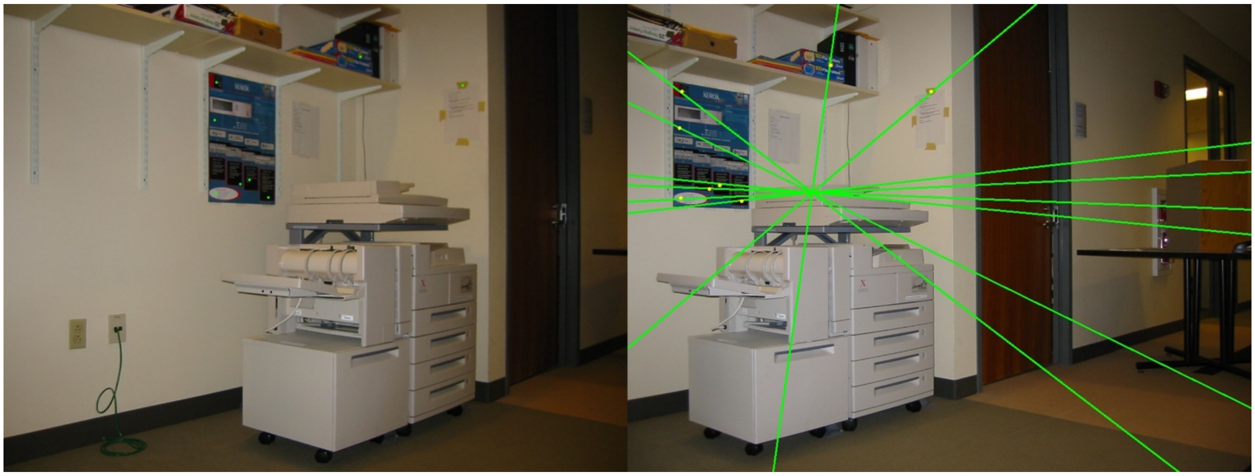
Question 1: Fundamental Matrix and Epipolar Line

Results:

Fundamental matrix:

```
[[ 4.74621161e-07  1.01762426e-05 -3.29429709e-03]
 [-9.17532975e-06  1.01184314e-06  2.46264337e-03]
 [ 2.58823452e-03 -5.75885878e-03  9.99971609e-01]]
```

Visualization of epipolar lines:



Question 2: Motion Field Estimation

Derivations of math at end

Results:

FOE: (3.0310, 4.0561)
Direction of translation: (0.3031, 0.4056)
Point 1 collision time: 105.0765
Point 2 collision time: 99.0195
Point 3 collision time: 97.3052
Point 4 collision time: 103.0278
Point 5 collision time: 104.2072
Point 6 collision time: 102.9162

Point 7 collision time: 101.8544
Point 8 collision time: 103.2691
Point 9 collision time: 96.2360
Point 10 collision time: 100.5332
Point 11 collision time: 99.1549
Point 12 collision time: 99.6377
Point 13 collision time: 100.1357
Point 14 collision time: 102.6670
Point 15 collision time: 100.2157
Point 16 collision time: 96.8932
Point 17 collision time: 99.2598
Point 18 collision time: 101.5265
Point 19 collision time: 102.0965
Point 20 collision time: 102.0703

With rotation correction:
FOE: (0.0014, 0.0018)
Direction of translation: (0.0001, 0.0002)
Point 1 collision time: 98.2846
Point 2 collision time: 100.1550
Point 3 collision time: 91.9262
Point 4 collision time: 100.1618
Point 5 collision time: 98.5952
Point 6 collision time: 103.8164
Point 7 collision time: 97.1929
Point 8 collision time: 98.2880
Point 9 collision time: 104.7728
Point 10 collision time: 69.3592
Point 11 collision time: 247.8206
Point 12 collision time: 100.6988
Point 13 collision time: 100.3456
Point 14 collision time: 100.9465
Point 15 collision time: 115.0705
Point 16 collision time: 90.2355
Point 17 collision time: 104.2101
Point 18 collision time: 95.9671
Point 19 collision time: 103.3990
Point 20 collision time: 98.8547

Question 3: Training a Neural Network on Fashion MNIST

1. Accuracy of baseline model on test set: *87%*
    a. *87.5%* accuracy on validation set
2. Modifications
    a. Smaller batch size of 32
        i. Accuracy on validation set: 87.4%

            ii. This hyperparameter controls the batch size for the gradient descent. A larger batch would result in larger jumps at every optimization step. I had thought that smaller batches would result in smaller jumps and thus a more gradual approach to the overall training. However, this model had slightly lower accuracy than the baseline on the validation set.
    b. Increased number of epochs (10)
        i. Accuracy on validation set: 88.3%
        ii. This hyperparameter controls how many rounds of training on the training set are performed on the model. This would likely improve performance, but may lead to overfitting. In this case, increasing the number of epochs increased the accuracy on the validation set.
    c. Lower learning rate of 0.001
        i. Accuracy on validation set: 85.9%
        ii. This hyperparameter controls the magnitude of change of the parameters at each optimization step. A very low learning rate results in the model not being able to adequately learn, while a very high learning rate might result in improper learning since the model will never smoothly reach a local minimum of the objective function. The chosen value (0.001) resulted in a lower accuracy on the validation set.
    d. Adding more channels to output of layer 1 (32 channels)
        i. Accuracy on validation set: 90.1%
        ii. This different construction of the first convolution layer controsl the number of features the model uses for classification. By increasing the number of channels in this output from 6 to 32, it increases the number of features the model can extract from the image. The result of this modification was an increase in validation accuracy, the best result so far.
    e. Adding dropout layer after first convolution layer, p = 0.3
        i. Accuracy on validation set: 88.7%
        ii. Adding the dropout layer is a form of regularization for the network, since some inputs will be 0 and thus will not contribute to the feed-forward prediction. As a result, neurons in the network must make up for each other and learn independently. The result of this addition is a slight increase in validation accuracy.
3. Best network: batch size of 64, 10 training epochs, learning rate of 0.01, 32-channel output of first convolutional layer, and dropout of p = 0.3 after first convolutional layer.
    a. Accuracy on test set: *88.0%*

Question 4: Fine-tuning a Pretrained Network on Caltech 101

1. Pre-trained network: Batch size of 64, learning rate of 0.005, 50 epochs
    a. Test set accuracy: 90.7%
2. Network from scratch: Batch size of 64, learning rate of 0.005, 100 epochs
    a. Test set accuracy: 73.2%

a. Focus of expansion (FOE) = (u,v) when $\frac{du}{dt}=0$ and $\frac{dv}{dt}=0$.

$w=0$

$$\to 0 = \frac{u_{foe}T_z - fT_x}{z} = \frac{T_z}{z}\left(u_{foe} - \frac{fT_x}{T_z}\right) \Rightarrow u_{foe} = \frac{fT_x}{T_z}$$

$$\to 0 = \frac{v_{foe}T_z - fT_y}{z} = \frac{T_z}{z}\left(v_{foe} - \frac{fT_y}{T_z}\right) \Rightarrow v_{foe} = \frac{fT_y}{T_z}$$

$$\to FOE = f \cdot \left(\frac{T_x}{T_z}, \frac{T_y}{T_z}\right)$$

$$\frac{du}{dt} = \frac{T_z}{z}\left(u - \frac{fT_x}{T_z}\right) \Rightarrow \frac{fT_x}{T_z} = -\left(\frac{z}{T_z}\left(\frac{du}{dt}\right) - u\right) = u - \frac{z}{T_z}\frac{du}{dt}$$

u_{foe} same for all points $\Rightarrow \frac{fT_x}{T_z}, \frac{fT_y}{T_z}$ constant

$$\frac{dv}{dt} = \frac{T_z}{z}\left(v - \frac{fT_y}{T_z}\right) \Rightarrow \frac{fT_y}{T_z} = v - \frac{z}{T_z}\frac{dv}{dt}$$

$$\frac{fT_{x,1}}{T_{z,1}} = \frac{fT_{x,2}}{T_{z,2}} \Rightarrow u_1 - \frac{z_1}{T_{z,1}}\frac{du_1}{dt} = u_2 - \frac{z_2}{T_{z,2}}\frac{du_2}{dt},$$

$$\hookrightarrow v_1 - \frac{z_1}{T_{z,1}}\frac{dv_1}{dt} = v_2 - \frac{z_2}{T_{z,2}}\frac{dv_2}{dt}$$

$$\frac{z_1}{T_{z,1}} = \frac{\left(u_1 - u_2 + \frac{z_2}{T_{z,2}}\frac{du_2}{dt}\right)}{\frac{du_1}{dt}}$$

$$\to v_1 - \frac{dv_1}{dt}\left(\frac{u_1 - u_2 + \frac{z_2}{T_{z,2}}\frac{du_2}{dt}}{\frac{du_1}{dt}}\right) = v_2 - \frac{z_2}{T_{z,2}}\frac{du_2}{dt}$$

$$-\frac{\frac{dv_1}{dt}}{\frac{du_1}{dt}}\left(u_1 - u_2 + \frac{z_2}{T_{z,2}}\frac{du_2}{dt}\right) = v_2 - v_1 - \frac{z_2}{T_{z,2}}\frac{du_2}{dt}$$

$$\frac{z_2}{T_{z_2}}\frac{du_2}{dt}\left(1-\frac{dv_1/dt}{du_1/dt}\right)=v_2-v_1+\frac{dv_1/dt}{du_1/dt}(u_1-u_2)$$

$$\frac{z_2}{T_{z_2}}\frac{du_2}{dt}=\frac{v_2-v_1+\frac{dv_1/dt}{du_1/dt}(u_1-u_2)}{1-\frac{dv_1/dt}{du_1/dt}}$$

$$\Rightarrow \boxed{\frac{z_2}{T_{z_2}}=\frac{1}{du_2/dt}\,\frac{v_2-v_1+\frac{dv_1/dt}{du_1/dt}(u_1-u_2)}{1-\frac{dv_1}{dt}/\frac{du_1}{dt}}}$$

$$\boxed{\frac{z_1}{T_{z_1}}=\frac{u_1-u_2+\frac{z_2}{T_{z_2}}\frac{du_2}{dt}}{\frac{du_1}{dt}}}$$

$$u_{foe,1}=u_1-\frac{z_1}{T_{z_1}}\frac{du_1}{dt}$$

$$u_{foe,2}=u_2-\frac{z_2}{T_{z_2}}\frac{du_2}{dt}$$

$\Rightarrow$ average for u_{foe}

$$v_{foe,1}=v_1-\frac{z_1}{T_{z_1}}\frac{dv_1}{dt}$$

$$v_{foe,2}=v_2-\frac{z_2}{T_{z_2}}\frac{dv_2}{dt}$$

average for v_{foe}

Direction of translation: $\left(\frac{T_x}{T_z},\frac{T_y}{T_z}\right)$

b). Time for collision $=\frac{z}{T_z}$