CM0699 Report - W14019187

Section A

1) **Give a brief description of the perceptual hashing technology and discuss its applications.**

A) Perpetual hashing allows for someone to hash data into a digital media item, such as an image or video using an algorithm. In short, it translates arbitrary data (usually binary) and into a fixed type that can be used to identify something uniquely as a fingerprint. In terms of application, you could use this to compare the legitimacy of two images. Each hash value when created by this algorithm should have a unique source and therefore any third-party modification that differs from the original algorithm can be easily detected. This can be useful for detecting if images or videos have been changed unknowingly.

2) **Explain how robust and fragile watermarking can be used to solve legal and ethical issues related to digital images and videos.**

A) The application of robust and fragile watermarking can solve several of these problems. Fragile watermarks at their basic are used for tamper detection and can detect if a source of media has been slightly modified or changed from it's original source. This type of watermarking is not "aggressive" and the original source will not resist being modified if someone were to modify an image with this type of protection. This could be used legally and ethically to compare two images or videos together to try an ascertain the original from the fake should it be modified without permission or some third party take credit for an image that they don't legally own.

Robust watermarking is similar but is much more aggressive and will stop a user from modifying the media it is embedded in. This type of watermarking is not too dissimilar to DRM as it stops the user outright from copying or modifying content without actual permission from the author. An application of this could be to protect intellectual property such as logos or videos which include an owners personal recordings as robust watermarking would outright deny easy access to manipulation or modification.

Section B

1) Discuss the main similarities and differences between Artificial Neural Networks and Decision Trees**.**

A) The first main difference is simply how each method of classification manages to train a set and then match it with the required output. For an Artificial Neural Network, it comprises it ‘self of “neurons” which represent a single linear classifier. These classifiers are then weighed on with information from an input source, once this happens the neuron reads the input data and decides if it is a match or not, this is decided in binary fashion (True or False) and data that passes a neuron moves onto the next. Generally, an ANN is a regressive form of classification, this means that it starts with multiple neurons nearer to the input stage and then ends with a single neuron for the output stage.

   In comparison, a Decision Tree follows a much different structure of doing the above process. Decisions tree feature an Internal Node, a branch and a Leaf for each section of the “Tree”. Internal Nodes a similar in a sense to a neuron in an ANN and provide the benchmark for the input data to be compared to. Branches are where data that does match the criteria is sent to and the leaf is the end result. Decision trees classify data recursively rather than regressively meaning that it sorts values in an attempt to create continuous values rather than calculating the values at each node. In addition, Decision Trees make use of entropy to try and make continues values more “genuine”. Entropy essentially adds a touch of randomness and authencity to possible outputs and sorting results. For instance, if sorting the weather for the day into Sunny, Cloudy or rainy, you would need some sort of randomness included as every day is very unlikely to be the same weather wise.

   The second point to consider is the speed at which each of these methods can be trained. Artificial Neural Networks are generally much slower than Decision trees once the initial training exercises have been completed, this is since DT’s sort information and “throw out” any that it deems to be too different to what it is trying to match it ‘self with whereas ANN’s keep all the input data regardless of similarity. This means that if you have a larger amount of data to sort through then a Decision Tree is most likely the better option whereas with smaller amounts of data a Neural network would be better. Another large difference is how each of these classification methods interpret new information added into the system after the initial training is completed. An ANN can easily interpret new input and incorporate them into what it already knows to help further the accuracy of its results. Decision trees on the other hand are calculated in “batches” and are much less dynamic than ANN’s. This means that usually when new inputs are received, it is best to simply retrain the entire data sets from the start to make sure that the data is incorporated better.

2) Explain why, in supervised machine learning, multiple samples from each class are required in the training phase.

A) Multiple samples are required for supervised machine learning for a number of reasons. Firstly, supervised learning requires that an end target or output be required at the end of the learning process, therefore multiple input samples are desired in order for a program to get a well-rounded result as we can use the expected output versus the multiple samples that are inputted. This is usually done via clustering (k) which sorts out these values into possible matches and possible non-matches. A second reason is in the nature of how supervised machine learning programs inherently work, that is they try and predict the outcome of the inputs through the entire training process. This works better with more inputs as it can compare multiple possible cases to build a better picture of how similar the images are. For instance, 5 pictures of different apples being inputted will make it easier to classify different types of apples better, rather than inputting one apple or 5 inputs of one single type of apple. By using multiple inputs in supervised learning it allows the system be better trained for any different types of what the output is trying to specify.

Section C

Code For Wavelet Transformation

A) Embedding

```
dm_wght = 0.3;
for i=1:8

   row=key(i,1);
   col=key(i,2);


     H2(row,col) = H2(row,col) + dm_wght *w1(i) * H2(row,col);
     V2(row,col) = V2(row,col) + dm_wght *w2(i) * V2(row,col);
     D2(row,col) = D2(row,col) + dm_wght *w3(i) * D2(row,col);

   c_A3=A2(row,col);

   Q_coe_A3=round(c_A3/Quantization_step);
   if watermark(i)==0,
      if mod(Q_coe_A3,2)~=0,
         Q_coe_A3=Q_coe_A3+1;
      end
   else
      if mod(Q_coe_A3,2)==0,
         Q_coe_A3=Q_coe_A3+1;
      end
   end
   reconstructed_coe_A2=Q_coe_A3*Quantization_step;
   A2(row,col)=reconstructed_coe_A2;
end
```

```
Reconstructed_A2=idwt2(A2,H2,V2,D2,'haar','mode','per') ;
Reconstructed_pepper=idwt2(Reconstructed_A2,H1,V1,D1,'haar','mode','per') ;
imshow(Image),title('Original peppers image');
figure,imshow(uint8(round(Reconstructed_pepper))), title('image with watermark')

imwrite(uint8(round(Reconstructed_pepper)),'wavelet_watermarked_image.tif');
```

Extracting Wavelet

```
Image=imread('wavelet_watermarked_image.tif');
w_H2=[];
w_V2=[];
w_D2=[];

dm_wght = 0.3;


for i=1:8

    row=key(i,1);
    col=key(i,2);

      H2(row,col) = H2(row,col) + dm_wght *w1(i) * H2(row,col);
      V2(row,col) = V2(row,col) + dm_wght *w2(i) * V2(row,col);
      D2(row,col) = D2(row,col) + dm_wght *w3(i) * D2(row,col);


   co_A3=A2(row,col);

   Q_coe_A3=round(co_A3/Quantization_step);
   if mod(Q_coe_A3,2)==0,
      w_H2=[w_H2,0];
   else
      w_H2=[w_H2,1];
   end
     if mod(Q_coe_A3,2)==0,
      w_V2=[w_V2,0];
   else
      w_V2=[w_V2,1];
     end
     if mod(Q_coe_A3,2)==0,
      w_D2=[w_D2,0];
   else
      w_D2=[w_D2,1];
   end
```

```
end

disp('extracted watermark at A3 (H2): '),w_H2
disp('extracted watermark at A3 (D2): '),w_D2
disp('extracted watermark at A3 (V2): '),w_V2
```

Manipulation Table + Explanation

	Wavelet-Based			DCT-Based		
Manipulation	W_1	W_2	W_3	W_1	W_2	W_3
Circular shifting [1,1] use 'circshift.m'						
Average filtering 7×7						

Explanation

Although I did not manage to obtain the codes I can offer a prediction on what would have happened should I have been able to complete it. I would assume that using a circular shift (circshift.m) would have shifted the bits of my watermark by one "row" and one "column" and shift the first bit in my watermark bit sequence to the end. The result would be something similar to
1 0 1 0 1 0 1 0 as per my code above if the shift was left.
Average filtering simply replaces all the values in an image or a matrix with an average. For instance, it would do a mean calculation for all the values in "peppers.tif" when extracted and calculate all the values to find a mean.

Code For Bays Training

Training

```
Bay1 = fitcnb(input_training_set,double(Target))
Disp(Bay1)
```

Testing

```
B_test = Bay1(Bay1,feature_vector);
```

Code for Perceptron Training

Training

```
P_test = newp(p_training_set,double(ptarget));
```

```
P_test = trainParam.epochs = 40;

P_test train(P_Test,p_training_set,double(ptarget));
```

Testing

```
p_testing = sim(P_test,feature,vector);
```

Code For Decision Tree

Training

```
D_tree=fitctree(input_training_set,double(D_Target))
view(D_tree,'Mode','graph')
```

Testing

```
D_testing_gender = predict(D_tree,feature_vector);
```

Classification Table + Explanation

	Linear Perceptron	ANN	Decision Tree	Naive Bayes
FPR	11/20 0.6	7/20 0.35	10/20 0.5	5/20 0.25
FNR	7/20 0.35	1/20 0.05	7/20 0.35	8/20 0.4
Error = $\frac{FPR+FNR}{2}$	0.42	0.20	0.425	0.35

NB: results are rounded when necessary.

5) **Analyse and interpret the results**

A)

The results for the use of an Artificial Neural Network show that the method is not one hundred percent accurate. It does however offer a reasonable total level of protection as it only managed to miss-classify 8 total images. We can find that by using an Artificial Neural Network that it is better at identifying males than it is females as seen by the lower FPR versus the FNR. The high level of accuracy compared to the other methods may be due to how ANN's use "neurons" to classify information much like a human brain. For the use of basic gender identification this is a very logical way to try and classify images of people as the human brain can easily classify genders of people with great accuracy.

The Linear Perceptron shows the worse results when running the code and offers an almost 50% failure rate. This is most likely attributed to the fact that this method of training is most suited to having a large number of inputs as it is generally required to obtain a reasonable level of accuracy. In this case it is likely that the lack of inputs has caused a large failure rate. Another possible consideration is the resolution and generated number vectors created by the program. As the resolution of some images is fairly small it would give a smaller vector for the program to work off of which may additionally cause problems when it comes to training and testing using this method.

The results for the decision tree are also fairly close to 50%. This is perhaps attributed to how this method of training handles inputs. It is likely that when classifying the values there may have been issues in the regression generation for the images that may have caused the program to badly classify both genders. As the decision tree hugely filters it's result it once again it relies on a higher volumes of results to work successfully.

Native Bayes offered a fairly average result and was moderately successful in classifying male and female images correctly. Unlike both Linear Perception and Decision Trees this method of classification runs more on probability based learning meaning that it captures areas of uncertainty and uses its own algorithm to try and solve classification questions with varying degrees of accuracy. In terms of this program the probabilistic nature combined with how native bays uses prior knowledge from it's training function has resulted in a reasonably accurate result.

6) **Explain how the performance of the gender classification system can be improved**

A)

There are a number of ways in which the classification system could be changed in order to make it better. For instance, some of the input samples could be better quality. Some of the gendered images have poor image quality that could hinder our algorithm to interpret the important gender based features that it needs to correctly identify each image, this could be facial structure or expression for example. In this case of this gender experiment, some images are blurry and out of focus which could cause this. A solution to this could be to increase the resolution of the image which would give more dimensions to use in the identification processes. For instance, a 512x512 image would have smaller dimensions to identify over an image that is 1028x1028.

Another important problem to consider is the fact that many people may have similarities between each other regardless of gender. For instance, many men have long hair which means that a classifier would have issues with identifying gender based on hair and other features such as a person wearing glasses may have some weighing on how similar gender-wise the algorithm may identify people. This could be improved by adding consideration for a person's hair into the program and other similar features. You could also perhaps identify the average facial structure of the average man or women, and base part of the classification on That, rather than weighing it more on facial features.

A final more general solution would be to simply train the program for a longer period of time. Longer identification and computation time spent on the image sis much more likely to produce a better result. Likewise, as this is a supervised programming program the use of more inputs may be a simple solution to training the program better as it has more data to classify against what is considered "male" and what is considered "female".