Chapter 1: Introduction

1.1 Project Overview

We implement two adversarial attacks on individual images on the image-net database the full data-set is not considered due to low computational resources as the data-set has more than three million images. We use a pretrained convolution neural network to identify the image. We use 1000 images for this task. The model is pre-trained on these images and has an accuracy of almost 99% on all images. But as we introduce subtle perturbations in the images the model beings to misclassify the images and starts predicting wrong class labels ,The difference between the unaffected and adversarial image is very difficult to spot and almost invisible to the naked eye(Fig(1)).To make the model robust to these attacks and prevent misclassification of true class labels by the model we use resampling techniques in the wavelet domain of the image and activation maps accompanied by pixel diversion to prevent misclassification of images in this dataset.

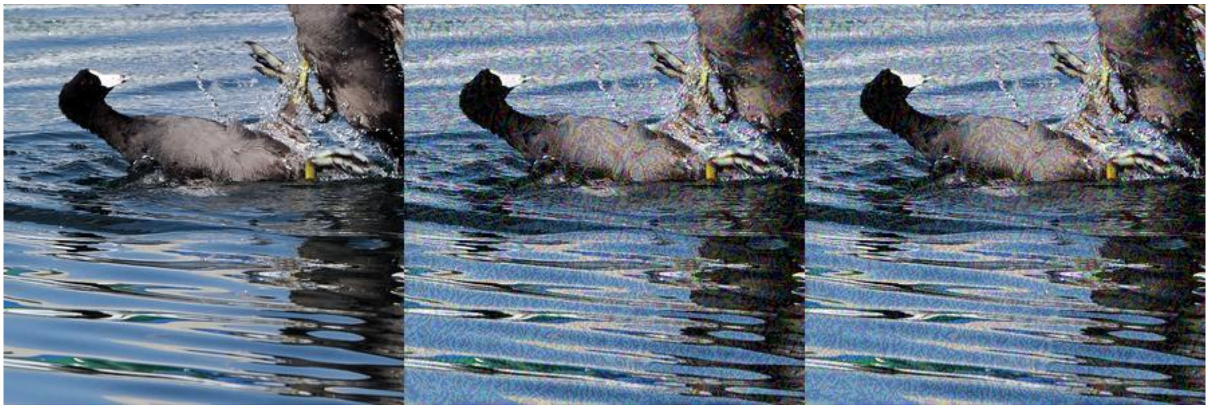


Fig (1)

(Leftmost is the original image and the other two images are affected with adversaries)

1.1.1 Technical Terminology

- Usage of python for implementing cleverhans library to import adversarial attacks (both targeted and non-targeted FGSM attacks)
- Using Scipy (Python library) to implement wavelet resampling to make classifier immune to adversarial examples.
- Fitting activation maps [18] which have been pre-trained and fitted on ImageNet samples. Which make the images perform well against adversarial attacks and help the classifier retain its true class label.

1.1.2 Problem Statement

Most current deep learning models are vulnerable to adversarial attacks/disturbances. An adversarial attack is a sample of an input image which has been modified in a way that is intended to cause a deep learning model to misclassify it. In most cases, these changes are subtle that even a human observer may not even notice the modifications at all, but these subtle changes will cause the model to misclassify the image thus causing harm and loss wherever technologies supporting these models are being used. Adversarial attacks pose security concerns because they could be used to perform an attack on computer vision models, even if the attacker has no access to the underlying algorithm. If implemented in the autonomous vehicle industry which relies on computer vision technology these subtle changes can lead to huge accidents leading to loss of life and damage to property.

1.1.3 Goal

Our goal in this project is to build an algorithm which is robust to these adversarial perturbations. So that even if an image has been compromised/attacked by an attacker using any adversarial attack the deep learning model still predicts the original/true class and its predictions are not swayed by the attack and it does not predict the wrong class/answer for the particular image.

1.1.4 Solution

The need of the hour is to build a state-of-the-art defence mechanism which would even help the model correctly classify an image even after it has been affected by adversaries. We have built an algorithm using randomized pixel diversion. We apply thresholding in the wavelet domain (wavelet denoising). Which has been known to work well with capturing distribution of pixels of natural images. This process smooths the adversarial images in such a way as to reduce effects of attack.

We also apply activation maps [18] to the images whose pixels we have randomly deflected. These class activation maps [18] replace the last CNN fully connected layer with a global average pooling layer. This results in a resulting map to be weak on pixel level precision but able to cluster objects by their actual class. These maps also help in masking the backgrounds of an image. Classification of classes often rely on co-occurrence in non-class objects. For example, Images of zebras often occur in lush wildlife and most classifiers will have low confidence when classifying zebra's out of this context. We use stochastic resampling of the background which keeps enough of the background for classification but drops enough pixels to weaken the adversarial attacks.

1.2 Need Analysis

As technology strives we become more and more dependent on machine's to do our work. The advancement of computer vision has helped us make face recognition system's in workplaces which prevent people to have their identity stolen, self-driving cars are being developed and deployed to reduce traffic and make travelling better recently Uber has launched its self-driving car program in Arizona. Even in the health sector it has been proven that image processing and computer vision techniques are being used to classify tumours and various types of cancers. S Thrun et al. [4] also developed a skin cancer classifier which has been able to out-perform dermatologists in detection of various types of skin cancers.

As we become more dependent on computer vision technologies we are more prone to adversaries that come with them. Neural networks are prone to adversarial examples [7, 8, 9] and any minor change in the classification of a neural network can make the classifier misclassify an image and even make the most accurate of DNN to predict wrong class labels.

If adversarial perturbations are added to autonomous vehicles [1,2] that can result in accidents which may even lead to loss of life. Propaganda can be spread through various videos which may be generated via deep-learning which are even impossible for humans to classify if they are original or not [4]. Medical sector has started to use predictions via DNN to classify and diagnose diseases [5,6]. Perturbation added to these images may even lead to loss of life.

To prevent this the need of the hour is to build a robust defence against these adversaries. An algorithm that doesn't let a DNN model misclassify an image even if it has been compromised.

We have built an algorithm which will help us defend prevent mis-classification of images even after the images have been affected by these attacks by using class activation maps and wavelet denoising which helps the deep neural network to work better against these adversaries and doesn't let it misclassify images.

Fig (2)

Self Driving Car

1.3 Research Gaps

Although many researchers have been able to create successful defenses from adversarial perturbations. There still is yet to be created a defence which is common for all kind of adversarial perturbations. Most defence models which use GAN's reject the image affected with adversarial perturbation [8]. Which leads to loss of important data for training. While models which use JPEG compression [2] as a way to defend images cause images to be misclassified when they aren't even affected by the attack, also there are constraints on when the particular defence is to be applied i.e. before or after an attack has taken place. These are some of the biggest gaps we've found in our analysis.

1.4 Problem Definition & Scope

The problem we have solved during the course of this project is to defend images from adversarial attacks.

Adversarial noise can be added to images which causes misclassification of images which can cause accidents (self-driving cars [1,2]) and even death (medical imagery [3]).

Our goal is to make a robust defence against these perturbations which does not let these perturbations mislead the classifier into misclassifying images.

The scope of this project is that this technology can be implemented in areas as fore-mentioned in computer vision technologies this algorithm can be applied in these industries to make the neural networks robust to adversarial noise. This will make the models less prone to misclassification saving accidents and misclassification in medical disease diagnosis.

1.5 Assumptions and Constraints

Table 1: Assumptions & Constraints

S. No.	Sample Assumptions
1	This is an algorithm which is used to defend DNN models from adversarial perturbations. What we have done in this project is as follows-: • Make a non-targeted adversarial attack that is used to sway the model away from the true class prediction. This model adds noise during back-propagation and subtracts noise to sway the model prediction away from the true class predictions [14]. • Making a targeted adversarial attack. This works opposite to that of the non-targeted example.

	In this attack noise is added to a prediction to make it follow and predict a particular class example [14]. • To deal with these perturbations we have designed an algorithm that does not let the model predict false value even if it gets affected with adversaries.
2	• Due to low computational power of our devices we have used the ImageNet database and the InceptionV3 pre-trained model to classify images and test accuracy on non-adversarial images as it gives almost 99% accuracy on the ImageNet database. • We also do not use the entire ImageNet database due to the database consisting of over 3 million images and we don’t have the computational power to process datasets this large, so we use 1000 images from the ImageNet database each of different categories.
3	For the facilitation of the attacks we have assumed that the model or classifier is a deep neural network as-: • In the case of both targeted and non-targeted attack both target the optimization/ weight updation of the model and is applied to the gradient descent step. • We have also assumed that these adversaries will take place on image data and no data of any other type. We have not taken into account video frames or text data. • We have also not taken account of RL algorithms.
4	• One other assumption about our research is that in the algorithm developed the attacker has no knowledge about the types of defences used and can access the information before and after the defence has been implemented so our algorithm works for cases when the defence is applied before the attack and even after the attack • One of the major assumptions about our algorithm is the development of robust-maps. we have used pre-developed maps created in [10] and made on ImageNet database. These maps would not be sufficient for image data that differs from the ImageNet database.

1.6 Approved Objectives

- Make one targeted adversarial attack.
- Make one non-targeted adversarial attack.
- Robust Defence mechanism to protect image misclassification against adversarial attacks.

1.7 Methodology Used

Our main purpose of this project is to build a robust model to defend from adversarial attacks. To test the ability of our defence we have built 2 adversarial attacks.

- Fast Gradient Step method (Non-targeted attack) -: The main idea behind this attack is to add weak noise on every step of back propagation in the neural network. Drifting the classifier from a true class to another class. We have to limit the amount of amplitude of noise added to keep the attack from being recognized by humans. This is a type of optimization. As in this case we optimize the error we have generated. In this case we can measure error and compute gradients as we have access to raw outputs of the network. It can be represented mathematically as -:

  - $x_0^{adv} = x , x_{t+1}^{adv} = x_t^{adv} + \alpha . sign(\nabla_x J(x_t^{adv}, y))$ [14]

- Targeted adversarial attack -: This works opposite to the non-targeted attack as mentioned above here we decrease the error and inverse the sign this makes the classifier shift to a particular class rather than predicting random outcomes.
  - $x^{adv} = x_t^{adv} - \alpha . sign(\nabla_x J(x_t^{adv}, y))$ [14]
  - This inverse sign reduces the error of the particular class we want to predict and shifts the classifier in the direction of that class.
- Defending against adversaries -: We have implemented 2 defences in our project and compared their outcomes and the accuracy they have achieved. First, we sample random pixels from an image and replace it with another randomly sampled pixel from a small neighbouring area so as to not disrupt the dynamics of the image. Then we implemented 2 forms of defence on these manipulated pixels -:
  - Maps -: We used class activation maps [18] to be put on top of images as to cover the image in view and some of the background area while discarding most of the area in the background. Classifiers are more robust to pixel diversion if pixels corresponding to the background are dropped. Luo et al. [10] used this idea in their research to mask regions which did not contain the object. This helps in the classifier only focusing on the image and not on the adversaries caused in the background. (Fig(3))
  - Wavelet resampling -: Wavelets localize features in our data to different scales. We can preserve important information while removing noise. The basic idea is that wavelet transform leads to sparse representation for many images and signals. Wavelet coefficients which are small in value mainly noise we can shrink these coefficients or remove them without affecting the quality of the signal. Pixel diversion and adversarial examples add noise to the image, so we apply a denoising transform to lessen the effect of the noise added. Wavelet coefficients are used. The transform represents the signal as linear product of orthonormal waves. Noise caused by dropping a pixel is high frequency but the same is not

true for adversarial noise in recent work by Luo et al. [10] has proved how most techniques fail to detect adversarial noise. Wavelet resampling outperforms these techniques.

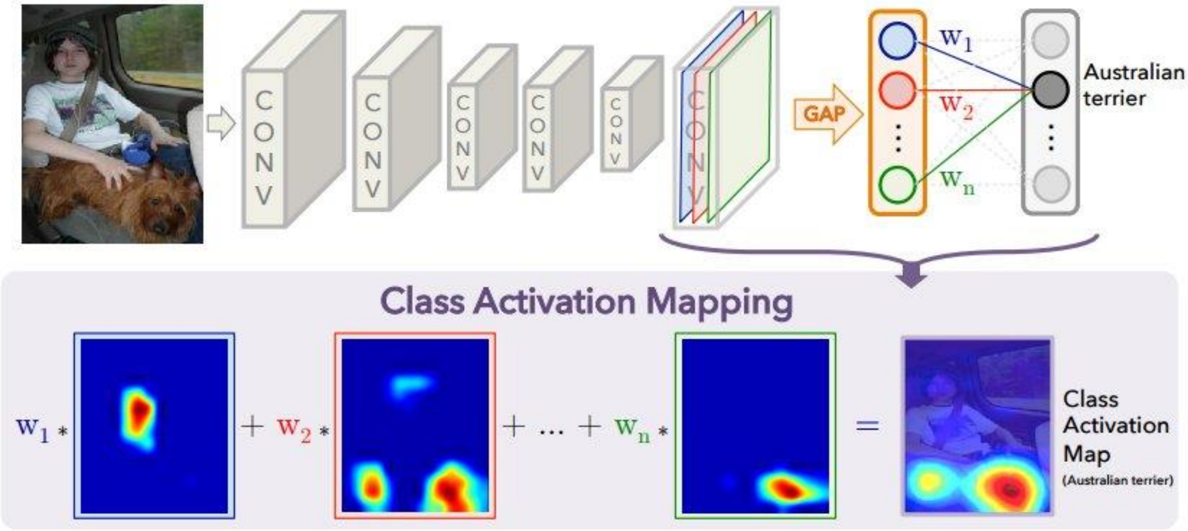


Fig (3)

Activation Maps applied with convolution neural network

1.8 Project Outcomes & Deliverables

During the course of this project we have made two adversarial attacks which can attack and influence any classifier to misclassify images. This can cause harm to many computer vision technologies that are being implemented in the industry. We have also developed a robust algorithm to protect the images from these adversarial perturbations and make the classifier perform well even if these attacks have happened on the image. Implementing these defences on images in which attacks haven’t taken place will not affect the classifier at all and will not change its prediction power but when any attack happens it still is confidently able to predict the true class of the object and does not get fooled by the perturbations. As the use of computer vision technologies are on a high from satellite mapping to autonomous driving industry to the medical sector. Using adversarial defence would not only help in curbing adversaries but also save the loss in terms of money and life that can be caused by the perturbations. Implementing the defence in accordance with the classifier as we have done in industrial projects will save money and curb the widespread of false information being spread.

1.9 Novelty of our work

We present a novel idea in implementing randomized pixel divulsion combined with activation maps [19] generated on the ImageNet dataset. This makes the classifier robust to adversarial noise.

Our method of using wavelet resampling makes us better than our counterparts who have implemented JPEG compression as JPEG compression causes misclassification of clean images. Our method is fast and uses less computational power.

Chapter 2: Requirement Analysis

2.1.1 Theory Associated with Problem Area

CNN’s have become an integral part of many system technologies. CNN’s are robust to any noise or data augmentation but pixels in an image can be manipulated by small, carefully designed and not visible perturbations, to make a perfectly accurate CNN model to misclassify images. With rapid progress and significance CNN’s and other deep learning models are used in the Medical industry [3,5,6], autonomous vehicle industry [7] and even used in offices for face detection and recognition.

However, these networks have recently been found vulnerable to well-designed input images altered by introducing adversaries in them. These adversarial examples are invisible to the human eye but can easily fool deep neural networks [8] when they are tested against images. This susceptibility to these adversarial perturbations make it very difficult to apply these deep learning frameworks in safety-critical environments.

Yen-Chen Lin et al. [18] stated how deep reinforcement learning agents are prone to adversarial perturbations. Using strategically timed attack [18] and enchanting attack [18] to minimize reward the authors were able to make the agent perform badly in its environment. Yen-Chen Lin et al. [18] showed how deep neural networks are vulnerable to physical attacks by applying a Disappearance attack [17] which caused a stop sign to disappear showing how this can be used to cause accidents in autonomous vehicles. They also showed the weakness of reinforcement learning algorithm Q-Learning against adversarial images. Huang et al. also proposed an attack on RL agent with adversarial examples that at every time step in an episode reduce the reward of the agent thus misguiding it. In terms of defending deep neural networks from these adversaries several approaches have been proposed Goodfellow et al. [8] augmented training data with adversaries to improve DNN’s to adversarial attacks by introducing generative model’s (GAN’s) which trained the generator to make fake images and the classifier to distinguish between real and fake images. Papernot et al. [12] proposed defensive distillation for training a neural network against FGSM attacks. Xie et al. [17] proposed to implement a stabling/measurement term in the objective function similar to generative models encouraging the DNN’s to generated similar adversarial outputs for adversarial versions of the image.

2.1.2 Existing System and Solutions

Goodfellow et al. [8] augmented training data with adversaries to improve DNN's to adversarial attacks by introducing generative model's (GAN's) which trained the generator to make fake images and the classifier to distinguish between real and fake images. This caused the generator and discriminator duel against each other and improve over iterations. Papernot et al. [12] proposed defensive distillation for training a neural network against FGSM attacks. Where same instances of two models are trained the first model is trained with the original data (X, y) where y indicates the class label and X is the input sample. The outcome probabilities are used to label the new dataset making it {x f(x)}) These new samples are trained on the second similar DNN. Xie et al. [17]. proposed to implement a stabling/measurement term in the objective function similar to generative models encouraging the DNN's to generated similar adversarial outputs for adversarial versions of the image. Other examples of defences in [1,2,3,7] have shown how original images exhibit similar pattern in their wavelet outcomes which could be used to learn from data and denoise the image. Swapping pixels in a window will add noise with unwanted frequency to the image. Although this noise can be removed by image compression techniques like JPEG, but this results in loss of original signal or features in an image. JPEG compression recovers images but also reduces the true class classification on natural images ([2]).

Our method is like that of Guo et al. [6] where the authors have used image processing techniques in which parts of the original image are replaced by similar parts drawn from a collection of images. It is augmented with variance minimalization in which another similar image is constructed, and noise is optimized. Variance minimalization is widely known for denoising images. Luo et al. [10]. also uses a defence mechanism in which the authors crop out the useful part of the image and scale it to the original size for the classifier so the chances of the DNN going wrong are minimal. Xie et al. [16]. also used a novel method of padding the image and taking random cropping from the image and evaluating the ensemble. This method takes advantage of the randomness of pixels in an image that we have been inspired to use in our algorithm.

Sibo Song et al. [14]. also integrated saak transform in their method of defending images from adversarial perturbations. Saak transform is a type of pre-processing technique used to defend images from adversaries. Saak transform is implemented in three steps firstly transforming an image to a spatial-spectral representation. Secondly filtering the high frequency components, and finally reconstructing the image via inverse Saak transform. The resultant image is said to be robust to perturbations.

2.1.3 Research Finding for Existing Literature

Table 2: Literature Review

S. No.	Roll Number	Name	Paper Title	Tools/Technology	Findings	Citation
1	101503174	Rohan	Defence Against Adversarial Attacks with Saak Transform	Python and Image processing techniques. Also uses Tensorflow.	It does not require any back-propagation training, no adversarial training and it is easy to implement. Also, it gives good classification accuracy on adversarial perturbations almost 47% on FGSM attack.	Song et al. [14]
2	101503174	Rohan	Adversarial Defence via Data Dependent Activation Function and Total Variation Minimization	Python and Tensorflow and cleverhans module	Data dependency function improves the classification of the DNN model and ability of the model to make correct guesses in the presence of adversarial examples. When combined with TVM a score of 20.6%, 50.7% and 68.7% is achieved w.r.t FGSM and I-FGSM attack methods	Wang, B et al. [16]

3	101503174	Rohan	Extending Defensive Distillation	Python	Defensive distillation trains two instances of a DNN. The first instance is trained upon the sample (x y) where x is the input images and y is the class label. The softmax classification output’s probabilities, these probabilities are used to label the training set x and now the input dataset becomes {(x f(x))} on which the second model is trained on this method is said to be prone to adversaries in images.	Papernot and McDaniel [12]
4	101503174	Rohan	Counting Adversarial Images using Input Transformations	Python and Image Processing techniques.	The paper implements 4 types of defence in this manuscript. It implements image quilting, TVM, JPEG compression and pixel quantization. The authors have used image processing techniques in which parts of the original image are replaced by similar parts	Guo et al. [6]

					drawn from a collection of images. It is augmented with variance minimalization in which another similar image is constructed, and noise is optimized. Variance minimalization is widely known for denoising images.	
5	101503174	Rohan	Foveation-based mechanisms alleviate adversarial examples	Python	In this manuscript the authors use a defence mechanism in which the authors crop out the useful part of the image and scale it to the original size for the classifier so the chances of the DNN going wrong are minimal. Even though adversaries are added, pixels are shuffled they remain in the same object frame hence the chance of going wrong are very less.	Luo et al. [10]
6	101503174	Rohan	Mitigating Adversarial effects through randomization		Xie et al also used a novel method of padding the image and taking random cropping from the image and evaluating the	Xie et al. [17]

					ensemble. This method takes advantage of the randomness of pixels in an image that we have been inspired to use in our algorithm.	
7	101503174	Rohan	Adaptive wavelet thresholding for image denoising and compression		S. Grace Chang et al. in their manuscript show that natural images can exhibit regularities in their wavelet domain which can be used to denoise that is cut out unwanted part of an image.	Chang et al. [1]
8	101562008	Sahil	Bayesian denoising of visual images in the wavelet domain.		The need for a measure of image distortion that adequately reflects human perceptual salience. Although the Bayesian denoising results in figure 15 are excellent according to a squared error measure, informal questioning suggests that most observers prefer a sharper image, even if it contains more noticeable artefacts.	Simoncelli [13]

9	101562008	Sahil	Ensemble adversarial training: Attacks and defences		Attacks crafted on adversarial models are found to be weaker even against undefended models (i.e., when using v3adv or IRv2adv as source, the attack transfers with lower probability). This confirms our intuition, adversarial training does not just overfit to perturbations that affect standard models, but actively degrades the linear approximation underlying the single-step attack.	Tramèr et al. [15]
10	101562008	Sahil	Keeping the bad guys out: Protecting and vaccinating deep learning with JPEG compression		Overall, we observe that applying JPEG compression (dashed lines with symbols) can counter FGSM and DeepFool attacks on the CIFAR-10 and GTSRB datasets. Φ means no compression has been applied.	Das N et al. [2]
11	101562008	Sahil	A study of the effect of JPG compression on adversarial images		By construction, JPG noise shares every permutation-invariant statistics with	Dziugaite et al. [3]

					JPG compression, but loses, e.g., information about the direction of the JPG compression modification	
12	101562008	Sahil	Relations between the statistics of natural images and the response properties of cortical cells		The response of a particular channel can be defined in terms of the variance of the filtered image. As noted earlier, for the rosette like codes the variance of the different channels will be roughly constant. However, a given variance can be produced by a range of different response distributions.	Field [5]
13	101503174	Rohan	Computer vision for autonomous vehicles	Computer Vision, Python, CNN	Using CV and sensors around a car a CNN is trained to identify lines on the road for training the vehicle. It is also trained on road sign and pictures of people using YOLO algorithm	Janai et al. [7]
14	101503174	Rohan	Dermatologist-level classification of skin cancer with deep neural	CNN, transfer learning ,Python	User interface for displaying internal state of autonomous driving system	Esteva et al. [4]

			networks			
15	101503174	Rohan	Learning deep features for discriminative localization.	Activation Maps, CNN	The authors target how the CNN visualizes objects and see images as it reaches the last layer. They implement a map over the last layer visualization to increase accuracy	Zhou et al. [19]
16	101503174	Rohan	Tactics of Adversarial Attack on Deep Reinforcement Learning Agents	Reinforcemt Learning ,Python,NVIDIA computing(cuda)	Two attacks on trained RL agents on advanced algos like DQN were implemented by the author.The first attack being the Strategically timed attack which attacked the agent in small intervals and minimized the reward and the second attack called Enchanting attack that lured the agent towards a particular action which caused mishaps in the game playing strategy of the agent.	Lin et al. [18]

2.1.4 The Problem That Has Been Identified

As seen and discussed in the literature above we have seen that DNN are prone to adversarial perturbations. We have also seen how the attacks manipulate pixel values during backpropagation [14] which causes a classifier to misclassify an image. Many research manuscripts have used JPEG compression for removal of adversaries from the image. This method has turned out to work well but it results in misclassification of the true image class when no noise has been added [14]. JPEG compression leads to the image losing its quality and surrounding making classifiers who sometimes make a prediction of what an object is according to its surrounding misclassify an image for example the picture of an airplane is often shown as clouds in the background. This removal of background from images causes misclassification of non-adversarial images. Many defence techniques have also proposed retraining the model after the images have been classified to be adversarial and corrected this increases the computation time and power for these defences. Papernot et al. [12] suggested the usage of two deep neural networks to be trained. The probability of true classes of the first model to be trained as the label for the other model, which causes increased computational power and increases training time. Other type of defences are based on machine learning theory involving GAN's and predicting whether an image is adversarial or not , Though this gives us a high accuracy [13,14] , It is not able to correct the flaws and adversaries in the image and that results in dropping a particular image which results in the training data to become smaller causing problems ranging from class imbalance to shortage of data.

2.1.5 The Survey of Tools and Technology Used

All authors in the literature have made use of python and Tensorflow due to the ease of use and the extensive libraries available in python for image processing. All the image processing algorithms are implemented in python and the back-end for the FGSM attacks and its different variations are implemented via cleverhans module which is a part of python library.

2.2 Standards

Following standard has been used for the development of this project: -

IEEE: The software requirement specification has been written using IEEE standards and also the references were written using IEEE standard.

IEEE802.11: The ESP8266 module supports this standard.

2.3 Software Requirement Specification

2.3.1 Introduction

Convolution Neural Networks have become an integral and definite part of computer vision technology. Despite augmentation being used to make better training examples, pixel values of an image can be manipulated, by small, crafted undetectable changes, to cause a model to misclassify an image. In our project we present a series of adversarial attacks to dislodge pixels by using Fast-Gradient-Step-Method using targeted and non-targeted methods [14]. We also present an algorithm to conserve the classifying accuracy when faced with these adversarial perturbations. With the help of this algorithm we aim to curb the various adversaries added to image classifiers. We also aim to encourage readers to get inspired by this method and explore this field of adversarial examples. So that they may be able to develop even better algorithms.

2.3.1.1 Purpose

The purpose of this project is to make a defence which is robust to adversarial perturbations. During the course of this project we aim to make DNN classifiers robust to adversarial perturbations such as a FGSM attack by implementing various algorithms. Such as wavelet resampling and using activation maps [19]

2.3.1.2 Intended Audience and Reading Suggestions

The intended audience for this project is the researchers who are interested in machine learning and image processing. This manuscript is also suggested for readers who are in the autonomous vehicle development industry and those who use deep neural networks and computer vision models in the health industry.

We would also like the audience to read this project keeping an open mind about the working of our defence model because the solution is not perfect and is an attempt to develop more interest of the readers in this field so that better results can be created.

2.3.1.3 Project Scope

The problem we have solved during the course of this project is to defend images from adversarial attacks. Adversarial noise can be added to images which causes misclassification of images which can cause accidents (self-driving cars [1,2]) and even death (medical imagery [3]). Our goal is to make a robust defence against these perturbations which does not let these perturbations mislead the classifier into misclassifying images. The scope of this project is that this technology can be implemented in areas as fore-mentioned in computer vision technologies this algorithm can be applied in these industries to make the neural networks robust to adversarial noise. This will make the models less prone to misclassification saving accidents and misclassification in medical disease diagnosis

2.3.2 Overall Description

Our project is based on the concept of defending images from perturbations. Images will first be affected by targeted and non-targeted FGSM attacks. Which would make the wrong/False class accuracy more prominent. Causing misclassification, to prevent misclassification we will use an algorithm divided into two parts 1.a) Wavelet resampling 1.b) Activation maps. This would make the accuracy of the true class back to being the highest.

2.3.2.1 Product Perspective

Our product comprises of three parts, two adversarial attacks one targeted and one targeted and one defence from these perturbations.

All are being implemented in Jupyter notebook environment. The activation maps used are downloaded from [19] as they were pretrained on adversarial images.

2.3.2.2 Product Features

No hardware need to be purchased for this product. It is a machine learning based applications which can be implemented in any industry in which it is found useful in.

2.3.3 External Interface Requirements

As mentioned above there is no as-such external interface requirement for our project it is a machine learning project which can be implemented in any industry in which it is found to be viable.

2.3.3.1 User Interfaces

We present all our algorithm and visualizations in Jupyter Notebook and we use no other form of GUI.

2.3.3.2 Hardware Interfaces

Although no hardware is required to implement our project it can be used in various industries as mentioned above with which it can be integrated to make it robust to perturbations.

2.3.3.3 Software Interfaces

We were not able to develop any GUI so visualization is done by using Jupyter Notebook which shows how the algorithm works.

- **2.3.4 Other Non-functional Requirements**
- **Usability**
  - The algorithm has all its visualizations done in Jupyter notebook which is user-friendly and good to view visualizations.
- **Performance**
  - The product promises a machine learning model that can classify fake or genuine images with high accuracy and confidence up-to 48 % or higher
- **Reliability**
  - The software will be able to efficiently make the classifier robust to adversaries introduced in it by the attacker.

- **Maintainability**
  - The software is flexible to modify and make changes according to the changing requirements that can be implemented while specifying, designing, coding and implementing provided the training set is not required to be altered.
- **Supportability**
  - After deployment the project will be a platform independent software that is portable as well as interoperable on other systems as well. Also it can be further used as a component in any other related software.
- **External requirement**
  - The product will not be able to disclose the identity of the person who performed the attacks on the image while on the network.

2.3.4.1 Performance Requirements

The main metric for evaluation for any machine learning project is the overall accuracy achieved. In this project we receive an overall accuracy of 48% on our best image.

2.4 Cost Analysis

Our algorithm uses open source software like Tensorflow and Keras for classification and as it is computationally inexpensive due to using a small subset of the ImageNet database we don’t require a GPU. Also, for making attacks the cleverhans module has been made open-source which doesn’t require any maintaining cost. Due to using open source frameworks there is no overall cost to make our project.

2.5 Risk Analysis

Like other frameworks which use JPEG compression for defending against adversarial examples which sometimes causes the classifier to misclassify on trained images. Our defence is robust to that and doesn’t affect the classification accuracy when no attack is applied to it.

Chapter 3: Methodology Adopted

3.1 Investigative Techniques

During the course of this project new scientific parameter and techniques are applied and discovered. FGSM[8] attacks are used to cause perturbations in an image these cause major misclassification by even a well trained and developed classifier to curb these adversaries from happening we have developed a new approach to defend images from adversarial perturbations.

The concept of activations maps[19] is derived from the knowledge of the area a convolutional neural network sees in its last layer before classification. For a trained model this area of image covered is less than the whole image. So by using this concept we mask our images with these activation maps making the area to add adversaries less and even if the pixels are now deflected they are in the vicinity of the object shape/ bounding border leading to less chances for misclassification.

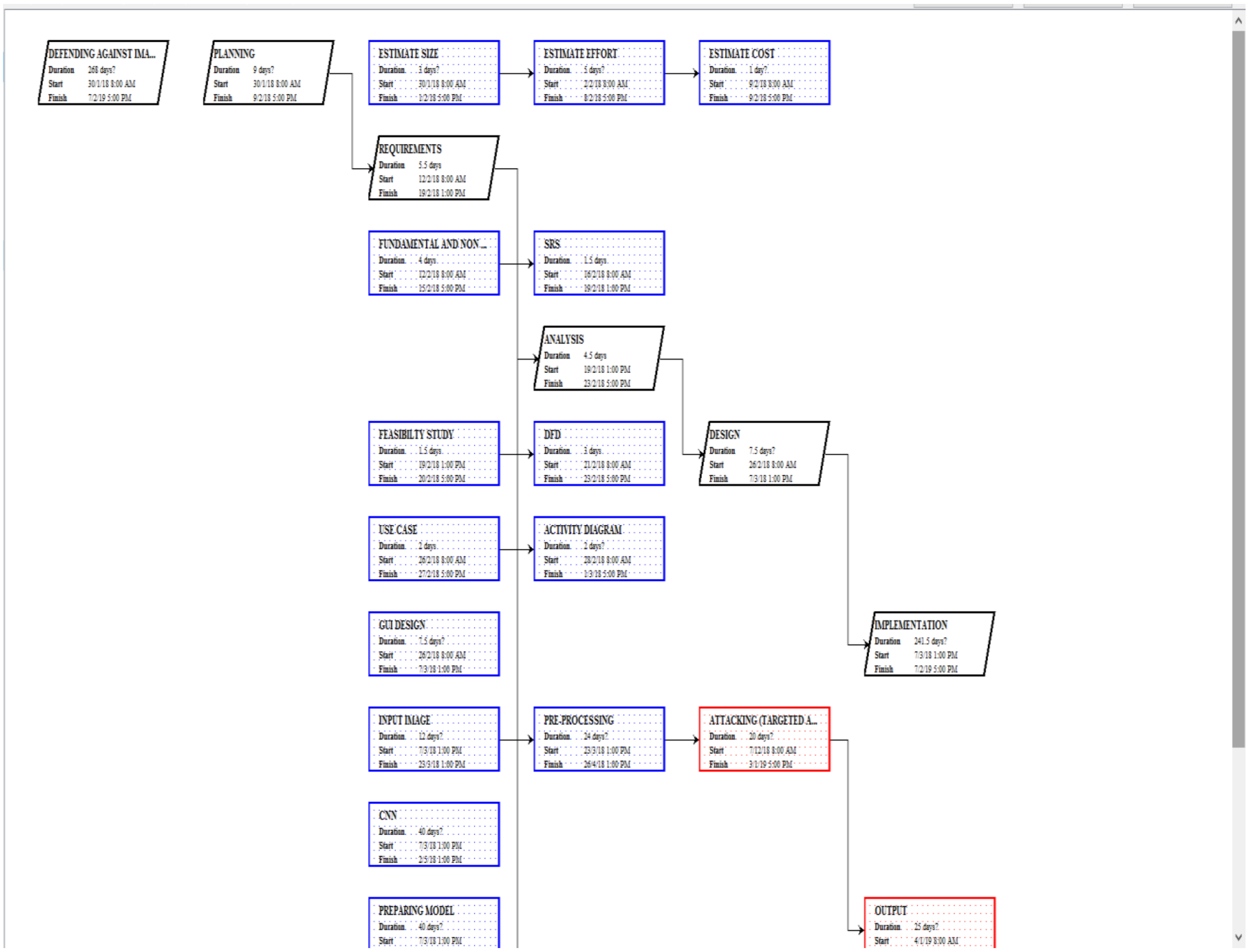
3.2 Proposed Solution

We propose two algorithms' as our proposed solutions to tackle the adversarial attacks faced by computer vision systems.

- Wavelet Denoising -: Wavelets localize features in our data to different scales. We can preserve important information while removing noise. The basic idea is that wavelet transform leads to sparse representation for many images and signals. Wavelet coefficients which are small in value mainly noise we can shrink these coefficients or remove them without affecting the quality of the signal. Pixel diversion and adversarial examples add noise to the image, so we apply a denoising transform to lessen the effect of the noise added. Wavelet coefficients are used. The transform represents the signal as linear product of orthonormal waves. Noise caused by dropping a pixel is high frequency but the same is not true for adversarial noise in recent work by Xie et al. [16] has proved how most techniques fail to detect adversarial noise. Wavelet denoising outperforms these techniques.

- Implementing Activation Maps-: Activation maps [19] are implemented on top of the image. There is one map generated for each particular class of the objects in the Image net database. That activation maps mark the major area of the image which is to be classified. Map are generally used for model to assign the maximum probability. However, when there is adversarial noise, the highest class is likely to become incorrect. We have observed in our experimentation that most perturbations affect the most likely class to be affected and do not change the other top x classes. We obtain a map which is robust to these fluctuations and we take the weighted average of the map on top x classes. We normalize the result to probability between [0-1], So that even if the top prediction is incorrect the averaging reduces impact of mis-localization of the image.

3.3 Work breakdown Structure



DEFENDING AGAINST IMA...
Duration 268 days?
Start 30/1/18 8:00 AM
Finish 7/2/19 5:00 PM

PLANNING
Duration 9 days?
Start 30/1/18 8:00 AM
Finish 9/2/18 5:00 PM

ESTIMATE SIZE
Duration 3 days?
Start 30/1/18 8:00 AM
Finish 1/2/18 5:00 PM

ESTIMATE EFFORT
Duration 5 days?
Start 2/2/18 8:00 AM
Finish 8/2/18 5:00 PM

ESTIMATE COST
Duration 1 day?
Start 9/2/18 8:00 AM
Finish 9/2/18 5:00 PM

REQUIREMENTS
Duration 5.5 days
Start 12/2/18 8:00 AM
Finish 19/2/18 1:00 PM

FUNDAMENTAL AND NON...
Duration 4 days
Start 12/2/18 8:00 AM
Finish 15/2/18 5:00 PM

SRS
Duration 1.5 days
Start 16/2/18 8:00 AM
Finish 19/2/18 1:00 PM

ANALYSIS
Duration 4.5 days
Start 19/2/18 1:00 PM
Finish 23/2/18 5:00 PM

FEASIBILTY STUDY
Duration 1.5 days
Start 19/2/18 1:00 PM
Finish 20/2/18 5:00 PM

DFD
Duration 3 days
Start 21/2/18 8:00 AM
Finish 23/2/18 5:00 PM

DESIGN
Duration 7.5 days?
Start 26/2/18 8:00 AM
Finish 7/3/18 1:00 PM

USE CASE
Duration 2 days
Start 26/2/18 8:00 AM
Finish 27/2/18 5:00 PM

ACTIVITY DIAGRAM
Duration 2 days?
Start 28/2/18 8:00 AM
Finish 1/3/18 5:00 PM

GUI DESIGN
Duration 7.5 days?
Start 26/2/18 8:00 AM
Finish 7/3/18 1:00 PM

IMPLEMENTATION
Duration 241.5 days?
Start 7/3/18 1:00 PM
Finish 7/2/19 5:00 PM

INPUT IMAGE
Duration 12 days?
Start 7/3/18 1:00 PM
Finish 23/3/18 1:00 PM

PRE-PROCESSING
Duration 24 days?
Start 23/3/18 1:00 PM
Finish 26/4/18 1:00 PM

ATTACKING (TARGETED A...
Duration 20 days?
Start 7/12/18 8:00 AM
Finish 3/1/19 5:00 PM

CNN
Duration 40 days?
Start 7/3/18 1:00 PM
Finish 2/5/18 1:00 PM

PREPARING MODEL
Duration 40 days?
Start 7/3/18 1:00 PM

OUTPUT
Duration 25 days?
Start 4/1/19 8:00 AM

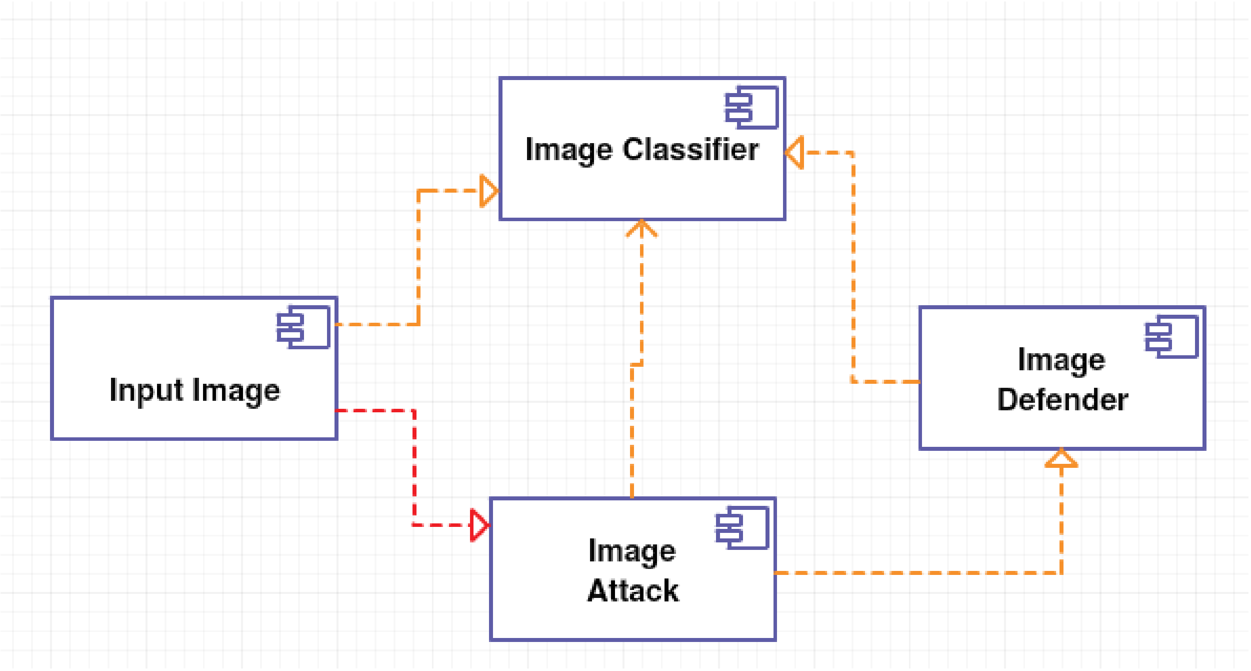
Fig(4) WBS

3.4 Technologies Used

- Python.
- Jupyter Notebook.
- Tensorflow.
- Keras.
- Scipy.
- Numpy
- Pandas
- Matplotlib
- Pretrained Resnet50 and InceptionV3
- Scikit-Learn
- CleverHans (module to develop attacks)
- Pillow

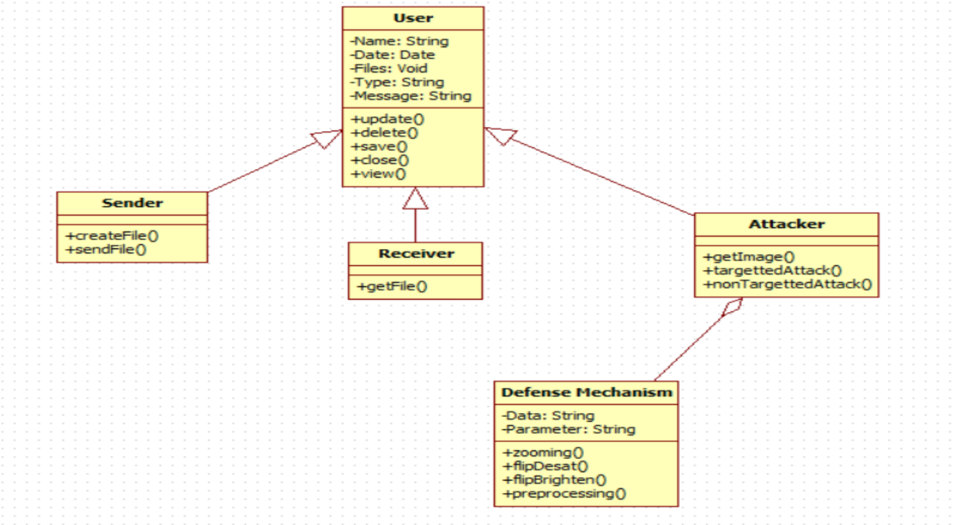
Chapter 4: DESIGN SPECIFICATIONS

4.1 System Architecture



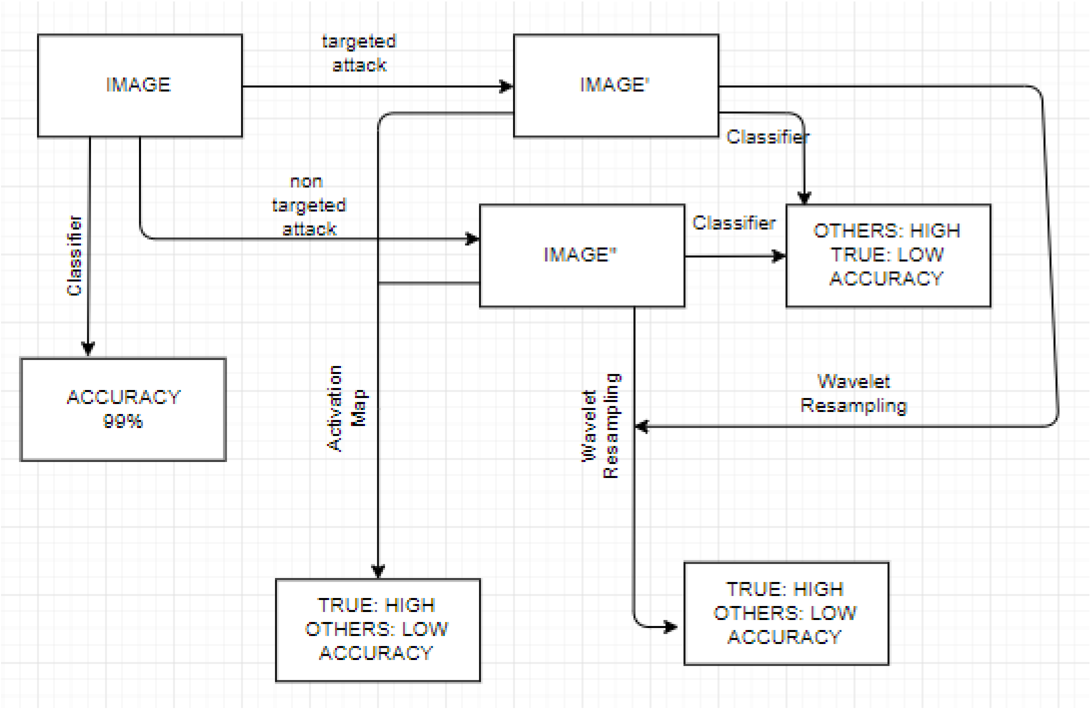
Fig(5) System Architecture Diagram

4.2 Design Level Diagrams



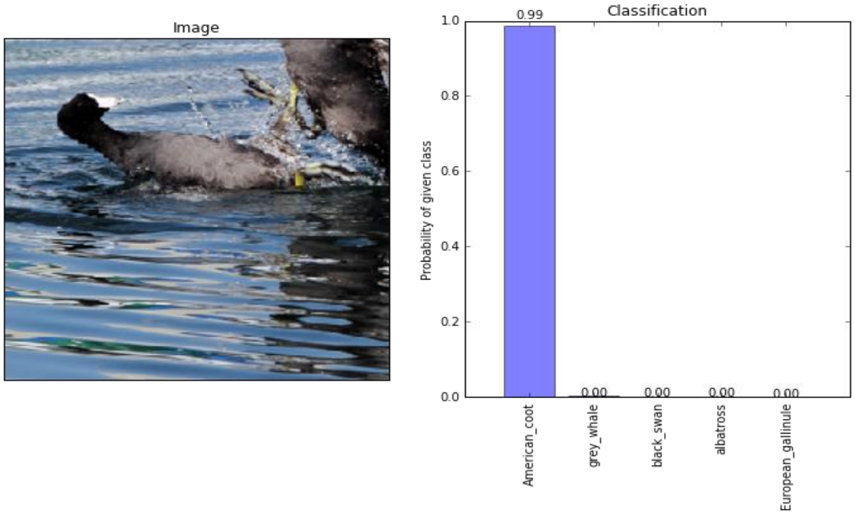
Fig(6) Design Level Diagram

4.3 User Interface Diagrams

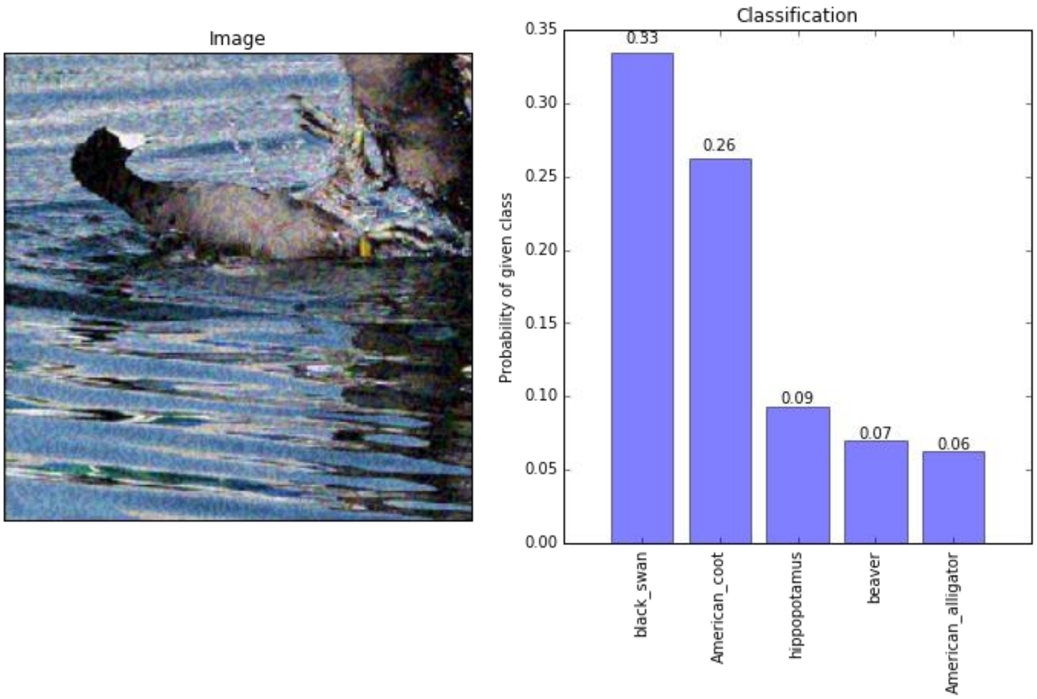


Fig(7) User Interface Diagram

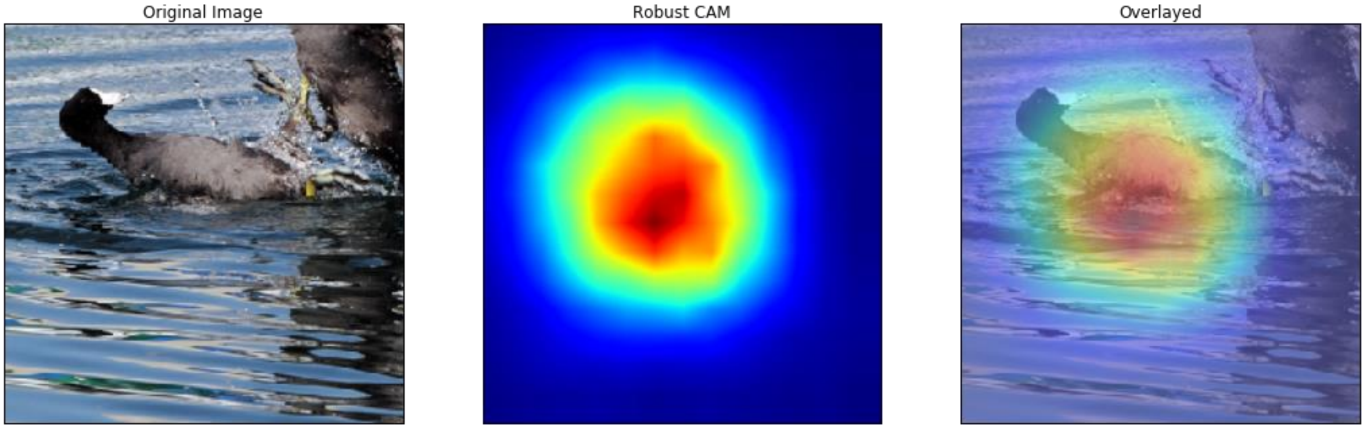
4.4 System Screenshots



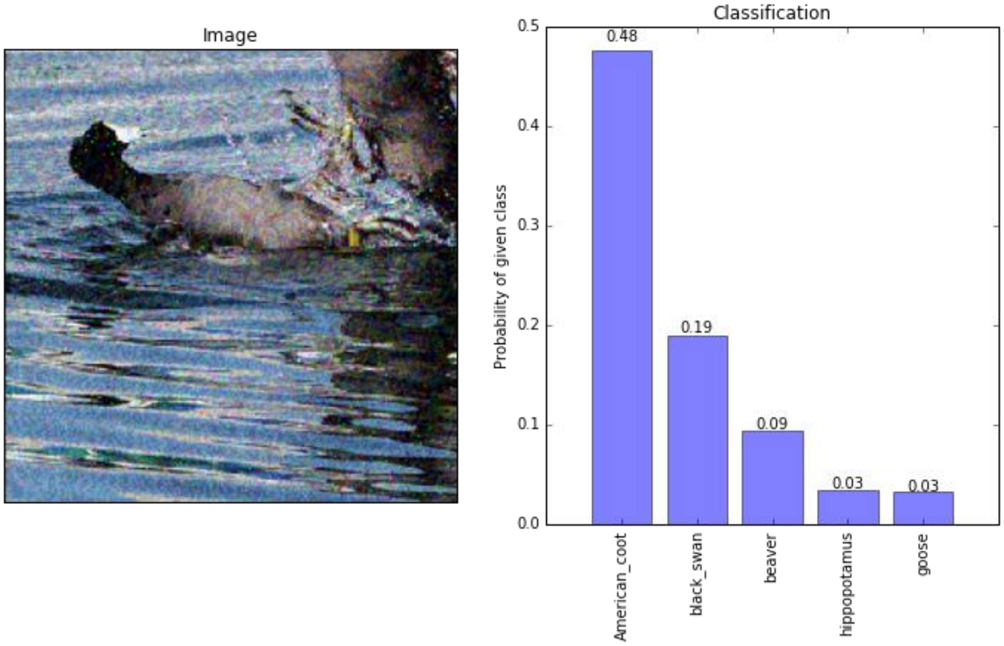
Fig(8) Original Image



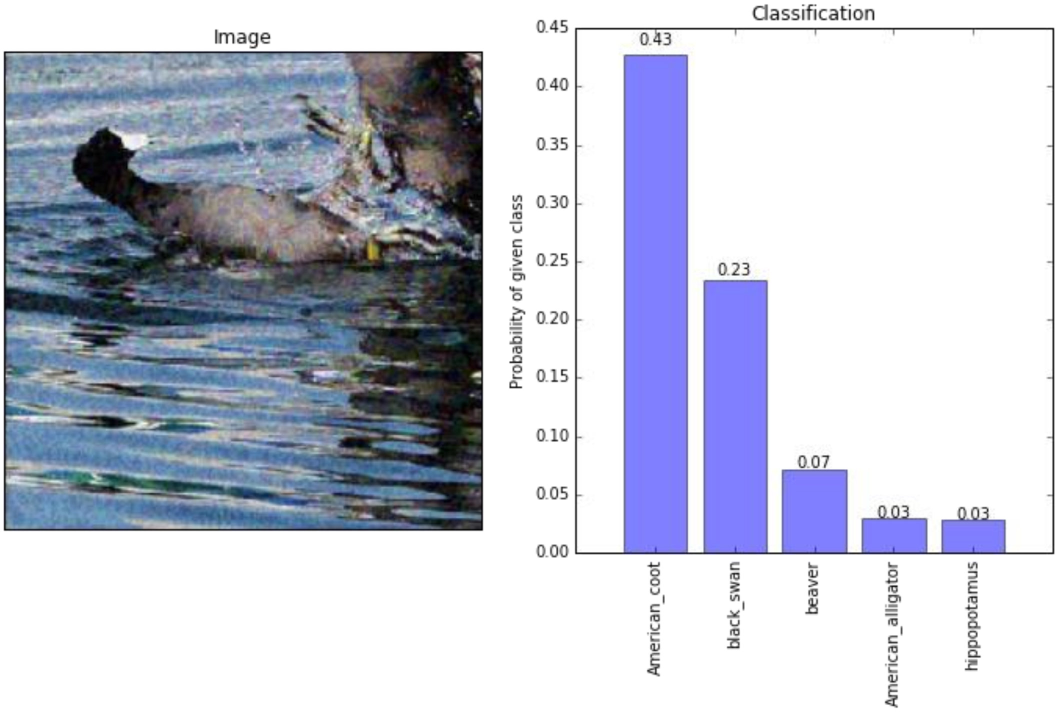
Fig(9) Adversarial Image



Fig(10) Picture Overlayed with activation map



Fig(11) Result after using activation map for classification



Fig(12) Result after using wavelet resampling on adversarial images

Chapter 5: Implementation and Experimental Results

5.1 Experimental Setup

A sample of 5 images were selected from the image-net dataset for running adversarial attacks and running the defence algorithm on them.

The images were picked randomly from the image-net dataset and were subjected to two types of adversarial attacks. Targeted and Non-targeted adversarial attacks which confused the classifier to classify the images as something which they were not. To overcome this we passed these adversarial images through the defence algorithm developed causing us to get an overall accuracy of 80%(4/5) on the validation set.

Before adversarial attacks the Inception model had an accuracy of almost 98-99% on all samples of image-net but after adding adversarial perturbations the second class/Non-dominant class had a higher accuracy than the major/true class. During recovery the model received an accuracy between 40-50% on each test image for the true class.

Due to less computational resources we were not able to run our attacks and defence on the whole Image-net dataset and had to stick to a lower number of validation samples.

5.2 Experimental Analysis

5.2.1 Data

The data for our project has been taken from the image-net dataset. Due to the size of the full dataset we've chosen a small subset of the dataset having 1000 categories and 1000 images. We've also chosen the InceptionV3 architecture for classification on the dataset. The dataset consists of a wide category of images ranging from dog breeds to buildings and castles.

5.2.2 Performance Parameters

We've only taken one performance parameter for the evaluation of our defence algorithm and that is the overall accuracy obtained by the algorithm on the dataset.

5.3 Testing Process

5.3.1 Test Plan

For testing we have taken five random images from the image-net database to perform testing for our algorithm

5.3.1.1 Features to be tested

There are three major features to be tested for this project. Which are as follows-:

- Non-targeted Adversarial Attack -: The first attack of the image suggests that the second dominant class becomes the major one causing disruption in the image and ultimately for the classifier to make wrong decisions/predictions.
- Targeted Adversarial Attack-: This is a variation of the non-targeted attack which only differs from the first approach on the basis of the gradient loss. So the correct working of this should be determined by implementing the attack on images and see whether the classifier misclassifies the images after the attack has taken place or not.
- Adversarial Defence-: This is the defensive algorithm we've built in this report. After the attack has taken place the aim of this algorithm is to make the classifier predict the correct class. We've used two approaches in our project a) wavelet denoising and b) Using activation maps. The main features to be tested here are the number of optimum deflections used in the denoising step, and whether the activation map has been correctly aligned on top of the image.

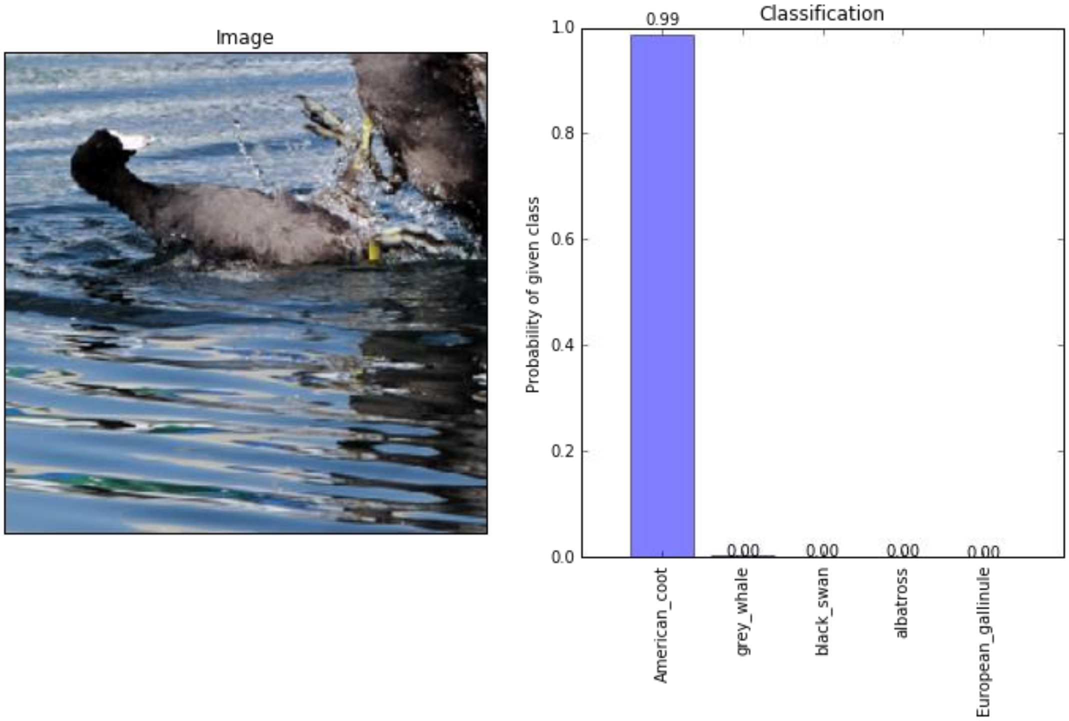
5.3.1.2 Test Strategy

Our test strategy lies on taking a non-manipulated image from the Image-net dataset and add adversarial perturbations to it by using the attacks we've developed above. When the images are attacked using the FGSM[8] the classifier predicts the class with the second highest classification accuracy to be the most dominant for example If the image of an American coot undergoes adversarial perturbation the second highest class ie the black swan becomes the most dominant class thus causing misclassification by the classifier As seen in the following figures.

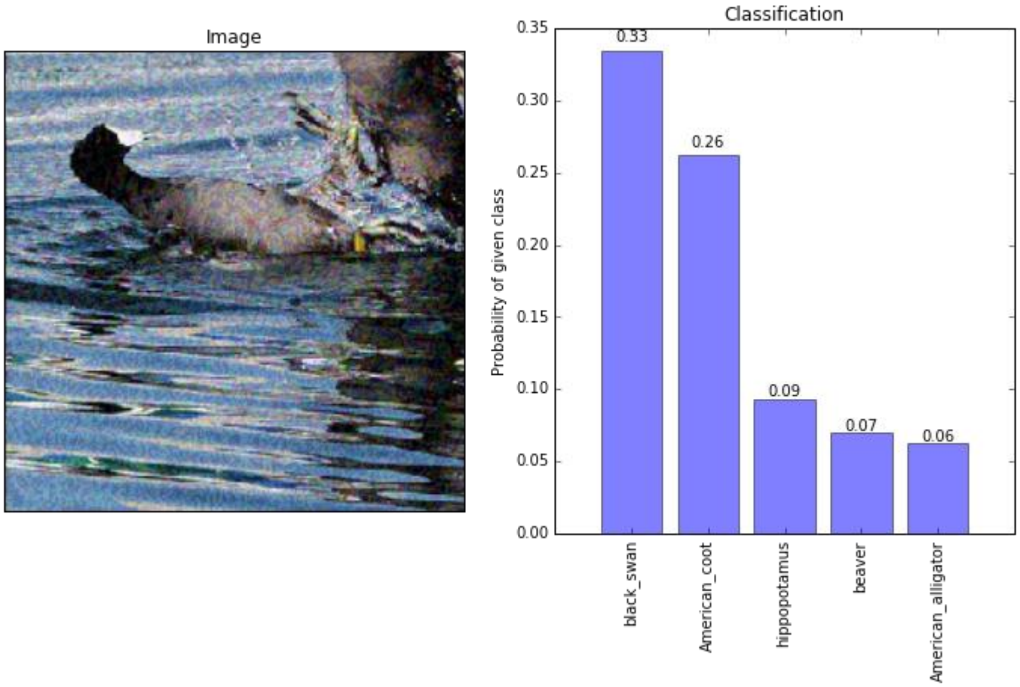
To test out the defence we want to leave out no test cases and exceptions that an attacker could take advantage of so we apply the defensive algorithm after the attack has taken place. Our aim with this defensive algorithm is that even after the attacks have been applied the algorithm still manages for the classifier to predict the true class with a good accuracy.

5.3.1.3 Test Techniques

As machine learning algorithms are prone to overfitting and generalize easily we have taken a wide range of images for testing. The categories of these images ranges from a simple American coot to a castle so that the defensive and the adversarial attacks are tested on a wide categories of images.



Fig(13) Original/Non-Manipulated image



Fig(14) Adversarial Image

5.3.2 Test Cases

The test cases for the dataset have been taken as a subset from the Image-net database, we take 5 images for validation due to less computational power of our machine. These images are passed through the attack function and their pixels are deflected causing misclassification by the classifier.

The clean images are passed through the attacks and then their final output is compared, the predicted class before and after should vary and that is true for most of the cases.

5.3.3 Test Results

We have presented the test results in the form of a table mentioned below.

We took random categories of images from the image-net database and passed them through the adversarial attacks and defence algorithm's The results are shown below.

Table 3-:Prediction results of the classifier after defence

S.NO	True Class Label	Targeted Adversarial Label Predicted	Non-Targeted Adversarial Label Predicted	Result After Applying Wavelet Denoising	Result After Applying Activation Maps
1.	American Coot (99%)	Black Swan (33% dominant class)	Jacamar	American Coot (47%)	American Coot (54%)
2.	Dung Beetle (99%)	Scale, Weighing Machine	Ground Beetle	Dung Beetle (43%)	Dung Beetle (47%)
3.	Giant Panda (99%)	Giant Panda (70%)	Giant Panda (66%)	Pomerium (43%)	Pomerium (70%)
4.	Sports Car	Sleeping Bag	Jet Engine	Sports Car (87%)	Sports Car (91%)

5.4 Result and Discussions

As we saw above the adversarial attacks implemented on the image work perfectly almost always making the classifier in predicting the false class. It showed how vulnerable convolution neural networks are to adversarial perturbations. We also see how these perturbations can harm convolution architectures implemented in the self-driving car industry. We have also seen how our baseline defence can help curb these perturbations. It can be further developed and enhanced in many ways including creating robust maps for each individual set of images causing better defence structures.

We also tested the algorithms on a small validation set to see its performance on a random set of images from image-net and it gave us an accuracy of 80%. The only downside of the defence is the low accuracy the classifier predicts of the true class after the attacks and defence have taken place ie Before the attack the accuracy of the classifier on image-net is around 99% but after the attack and defence have been applied the class predicted is correct but the accuracy drops down between 40-50%, Improvements can be made in this field by exploring new ways and algorithms and further optimizing existing steps.

We also observe something very interesting when testing the algorithm sometimes the adversarial attack can't harm the image to such an extension that the major class is misclassified. When this image is passed through the defence algorithm it can be seen that after defence it's accuracy of the major class drops than that it was before the attack.

5.5 Inferences Drawn

After testing and working with our attacks and defensive algorithm's we have concluded that convolution neural networks are prone to adversarial perturbations[4,7] and this can lead to a lot of problems in industries which use machine learning and AI for growth. We have also seen how methods to curb these perturbations are actively being created and deployed,

We've also shown our defensive strategy and how it can be further enhanced as mentioned above to avoid these perturbations to cause damage to existing neural frameworks. Many defences are being developed and deployed and activation maps[19] gives these defensive architectures an edge as they use the theoretical idea of how a convolution neural network views an image and by restricting the reach of the classifier to that box we can prevent perturbations causing high level of damage to images as they restrict the viewing area of the network/classifier leaving only the required image in the bounding box/area.

The dropping of classification accuracy on an image that isn't affected by adversarial attacks decreases when it is passed through the defensive algorithm showing that the approach is not robust and gives the best results when the image passed is manipulated to a certain level.

5.6 Validation of Objectives

The algorithm developed above gives an 80% accuracy on validation set against adversarial attacks, although this algorithm is limited to the image-net database with the right resources and classifiers this can be expanded to industry use.

The attacks/ perturbations are intensively checked against images of the image-net dataset and are easily able to fool the classifier. They also are not visible to the naked eye making any human observing the attacks obsolete.

The defence algorithm also works well and gives an overall score of 80% on validation set on image-net database.

Our project is written in Tensorflow and is susceptible to changes/updates making the old syntax obsolete, so it is susceptible to changes overtime and has to be updated in regular intervals.

To test the defence algorithm the images are passed through the various attacks developed and then passed through the defence algorithm to test whether it is giving the correct prediction or not.

Table 4-:Validation of Objectives

Sno	**Task**	**Result**
1.	Non-Targeted Adversarial Attack	Successful
2.	Targeted Adversarial Attack	Succesful
3.	Defense Algorithm	Succesful

Chapter 6 : Conclusions and Future Directions

6.1 Conclusion

During the course of this project we draw many conclusions by the form of defence used against adversarial perturbations. Seeing the nature of CNN's against even the basic of adversarial attacks we see how adding even a bit of noise can change the whole prediction of the classifier. We presented an algorithm which combines image transformation, pixel diversion with wavelet denoising. This technique provides a moderate defence against strong adversarial attacks. Most attacks are sceptic to well-formed content and using pixel diversion with normalized activation maps protects important regions of an image. We also show a robust method of defending DNN's from adversaries by using minimal computational power which makes our method cost effective, easy to use and easy to train and get results quickly. We tested our defence against two types of FGSM attacks, but we encourage users to test this against more adversarial attacks so that this algorithm could we further improved. We were able to achieve a mean accuracy of 45% on correct classification of adversarial images using our normalized activation maps and a mean accuracy of 42% on using wavelet denoising with random pixel deflection. Our algorithm is a starting step for further implementation and growing in this area of defending images from adversaries. As discussed above adversarial examples also affect RL algorithms which can cause a stand-still in research for AI algorithms. We can use and implement learning's from this project to see how to solve problems which cause adversaries in AI.

6.2 Environmental, Economic and Societal Benefits

As stated due to the gaining popularity of computer vision techniques and every part of the industry defending images from adversarial attacks has become very important.

Image classifiers are outperforming humans on many levels, Nowadays algorithms have been built that can classify and diagnose various medical diseases[4] better than a average physician which cost less than 1/4th of the total cost saving people money and helping in better diagnosing of diseases. The use of these classifiers doesn't stop here they are also being used in autonomous vehicle industries[7]. These vision technologies are used to build robots for defence. Recently a fake video of American president Donald Trump was released talking propaganda which turned out to be fake and a result of adversary caused in an image using GAN's and targeted-adversarial noise. This shows that if adversaries are let loose in this field it could result in widespread propaganda.

As we see above CV and DNN's are taking over the technological world, now if these classifiers are compromised this can affect lives of millions of people. To avoid this, we have developed an algorithm that helps the classifier to predict the true class label even if the images have been compromised. This can save millions of dollars of damages if implemented with autonomous vehicle technology and can be a deciding factor between life and death in medical imagery.

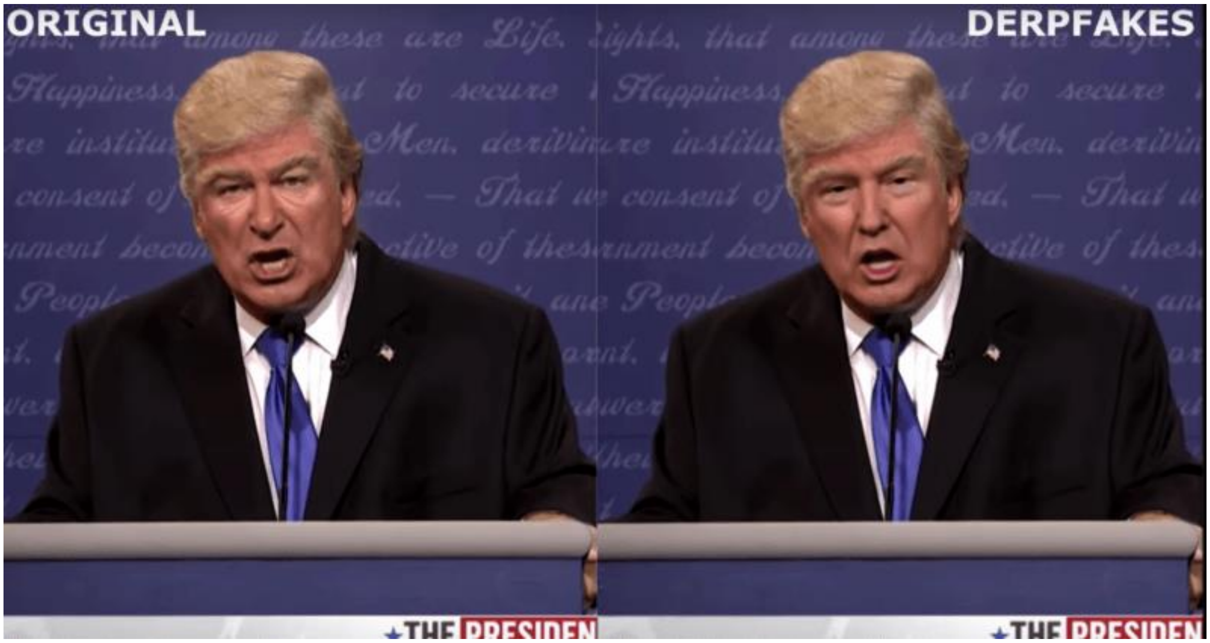


Fig (15) Derpfakes(Real vs Fake Trump)

6.3 Reflections

During the course of this project we came learnt a lot about how computer vision technologies work and how easily they can be corrupted. During the learning phase we learnt how to build and run adversarial perturbations and use them to fool a model into misclassifying images. We developed two targeted attack's which when run on the ImageNet dataset gave good results. We also observed how less explored is the field of adversarial examples and defences. We observed the dangers of adversaries and came across DeepFakes (web article) an algorithm which has been recently developed and deployed and is creating a lot of buzz in the field of CV. It has been successfully able to copy one person's movement and actions to another and is capable of even fooling a human observer as seen in Fig(17) below. These attacks if deployed would lead to a lot of hardships and loss suffered by the people.

In the literature we came across many methods for the defence which were restricted to certain constraints and could not perform if the certain assumptions weren’t satisfied. We also observed the research gap b/w building a solid defence and attack and how perturbations are easily built and deployed which cause mayhem while building a defence is dependent upon many constraints and there still hasn’t been a defence that would work well against all types of adversaries and noise.

Building of a solid defence is still a long way to go but with our algorithm we hope other researchers would find it useful and develop it further so as it can help the society.

Fig (16) Obama(Real vs Fake)

6.4 Future Work

In the current work we would like to improve our technique of evaluating suitable hyperparameters for individual images and be able to extract and develop accurate hyperparameters for each image. We also would like to prepare an ensemble of activation maps with wavelet denoising to see how the combination of two of our defences work together.

Furthermore, developing of activation maps for not only the ImageNet database but for every image would also help in increasing the accuracy by a large margin.

Chapter 7: Project Metrics

7.1 Challenges Faced

During the course of this project we faced many challenges. Starting with the learning of new concepts to try and implement this task. Our first major challenge was the development of the adversarial attacks. Learning tensorflow and then ultimately pytorch(an easier platform for implementing DL algorithms) was tough at the beginning but woth the help of MOOC's we were able to overcome this hurdle. The logic for the algorithm for FGSM was inspired by this paper[6]. The second challenging task was the building of the defense algorithm going through the literature and first trying to see how the recent algorithms work. We also experimented with basic defensive algorithms but they were of no use, So we applied activation maps [19] to the image for the defensive approach and it worked the best among all the approaches we tried previously.

7.2 Relevant Subjects

Table 5 : Subject Code and Subject Name

Subject code	Subject name	Description
UCS 602	Machine Learning	Using ML techniques for defence algorithm
UCS615	Image Processing	Manipulating images and using it to oversample the training set
UCS503	Software Engineering	Using software engineering to make useful architecture diagram
UCS802	Deep Learning	For building classifiers and attacks

7.3 Interdisciplinary Knowledge Sharing

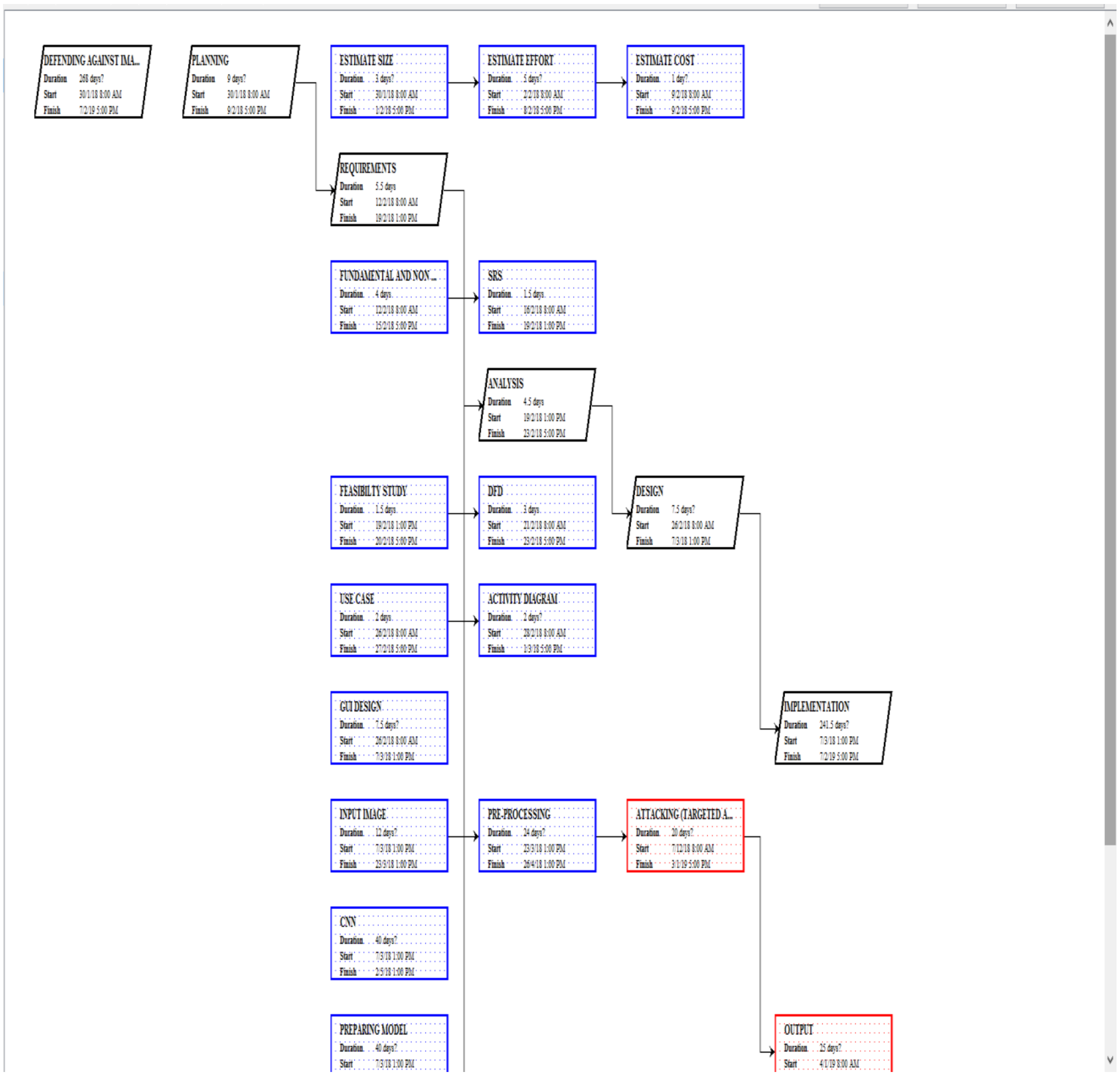
There were difficulties in deciding what to work on, because of the mismatch between the technical skills and quality of the team and good research problems can be found only in the interval between trivial problems and intractable ones. So, to solve this problem we shared our knowledge with each other. This helps the one which requires a solution incorporating complex technical aspects from more than one area.

7.4 Peer Matrix Table

Table 6-:Peer Assessment Matrix

		Evaluation By	
		Sahil	Rohan
Evaluation By	Sahil	4	5
	Rohan	5	4

7.5 Role Playing and Schedule



DEFENDING AGAINST IMA...
Duration 268 days?
Start 30/1/18 8:00 AM
Finish 7/2/19 5:00 PM

PLANNING
Duration 9 days?
Start 30/1/18 8:00 AM
Finish 9/2/18 5:00 PM

ESTIMATE SIZE
Duration 3 days?
Start 30/1/18 8:00 AM
Finish 1/2/18 5:00 PM

ESTIMATE EFFORT
Duration 5 days?
Start 2/2/18 8:00 AM
Finish 8/2/18 5:00 PM

ESTIMATE COST
Duration 1 day?
Start 9/2/18 8:00 AM
Finish 9/2/18 5:00 PM

REQUIREMENTS
Duration 5.5 days
Start 12/2/18 8:00 AM
Finish 19/2/18 1:00 PM

FUNDAMENTAL AND NON...
Duration 4 days
Start 12/2/18 8:00 AM
Finish 15/2/18 5:00 PM

SRS
Duration 1.5 days
Start 16/2/18 8:00 AM
Finish 19/2/18 1:00 PM

ANALYSIS
Duration 4.5 days
Start 19/2/18 1:00 PM
Finish 23/2/18 5:00 PM

FEASIBILTY STUDY
Duration 1.5 days
Start 19/2/18 1:00 PM
Finish 20/2/18 5:00 PM

DFD
Duration 3 days
Start 21/2/18 8:00 AM
Finish 23/2/18 5:00 PM

DESIGN
Duration 7.5 days?
Start 26/2/18 8:00 AM
Finish 7/3/18 1:00 PM

USE CASE
Duration 2 days
Start 26/2/18 8:00 AM
Finish 27/2/18 5:00 PM

ACTIVITY DIAGRAM
Duration 2 days?
Start 28/2/18 8:00 AM
Finish 1/3/18 5:00 PM

GUI DESIGN
Duration 7.5 days?
Start 26/2/18 8:00 AM
Finish 7/3/18 1:00 PM

IMPLEMENTATION
Duration 241.5 days?
Start 7/3/18 1:00 PM
Finish 7/2/19 5:00 PM

INPUT IMAGE
Duration 12 days?
Start 7/3/18 1:00 PM
Finish 23/3/18 1:00 PM

PRE-PROCESSING
Duration 24 days?
Start 23/3/18 1:00 PM
Finish 26/4/18 1:00 PM

ATTACKING (TARGETED A...
Duration 20 days?
Start 7/12/18 8:00 AM
Finish 3/1/19 5:00 PM

CNN
Duration 40 days?
Start 7/3/18 1:00 PM
Finish 2/5/18 1:00 PM

PREPARING MODEL
Duration 40 days?
Start 7/3/18 1:00 PM

OUTPUT
Duration 25 days?
Start 4/1/19 8:00 AM

Fig(17) WBS

7.6 Student Outcomes Description and Performance Indicators (A-K Mapping)

Table 7-:A-K mapping

SO	Description	Outcome
A1	Applying mathematical concepts to obtain analytical and numerical solutions.	Yes we used the concept of gradient descent during back propagation to manipulate images to apply adversarial perturbations.
A2	Applying basic principles of science towards solving engineering problems.	As the research in ML grows it becomes more prone to hackers, We use knowledge gathered over our vast literature to come up with ideas to avoid these hackers from damaging images and causing havoc.
B1	Identify the constraints, assumptions and models for the problems.	During the course of the project many constraints regarding the defense algorithm were added. The major one being the usage of activation maps works best with image-et data and for other images activations maps need to be developed.
C1	Design software system to address desired needs in different problem domains.	This software can help computer vision companies all over the world from medical imaging to autonomous vehicle.
D2	Can play different roles as a team player.	All of us in the team contributed equally and played all roles from coding to documenting our research to make this project viable.
E2	Develop appropriate models to formulate solutions.	We have developed activation maps to develop an aaccurate and good solution for the problem faced in this report
F1	Showcase professional responsibility while interacting with peers and professional communities.	As professionals we divied our work and helped each other out whenever someone needed help. Both our team members were involved in all aspects oof the project from coding to documentation. We also interacted with our respected mentor on regular basis and shared our findings with her
F2	Able to evaluate the ethical dimensions of a problem.	Our solution is based on the

		ethics violated by the attackers to cause disturbance in an image. We were able to evalulate the ethics of the problem
H1	Aware of environmental and societal impact of engineering solutions.	We are well awar of the environmental and social impact of the solution. We have made our project such as it is compatible with all resources and can be merged with any CV domain
H2	Examine economic tradeoffs in computing systems.	The major economic trade off is the cost of building the system and the performance. Our project is made free of cost using open source libraries and gives good results hence it is economically viable.
I1	Able to explore and utilize resources to enhance self-learning.	During the course of this project we learnt a lot of new technologies on our own by taking part in online MOOC's and self reading books.
I2	Recognize the importance of life-long learning.	During the course of the project we've learned a lot about python,machine learning and deep learning these skills will certainly help us in our future endevours
J1	Comprehend the importance of contemporary issues.	The project is made keeping in mind the contemporaries in computer vision. We use this project as a baseline foundation for our future work in defending images from adversaries.
K1	Write code in different programming languages.	We have written all of our code in python and tensorflow. We have even used pytorch for quicer results.

7.7 Brief Analytical Assessment

Through the course of this project we gathered most of our information from research papers and the deep learning book written by Ian Goodfellow.

The aim of our project was to develop a robust algorithm to defend images from adversarial perturbations. Our current implementation is a carry-forward of the idea of how neural classifiers see images and how that area can be exploited and used as a technique for defending images from adversaries

The project is based on applied machine learning techniques and forced us to think of new alternatives to solve the problems faced in the project. Using the mathematical knowledge of pixel deflections and computing knowledge of how classifiers see objects we were able to come up with our solution.

Our team members were of different branches and we had to work around our schedules and meet regularly outside college hours to make this project possible.

References

[1] Chang, S.G., Yu, B. and Vetterli, M, "Adaptive wavelet thresholding for image denoising and compression." *IEEE transactions on image processing*, *9*(9), pp.1532-1546. 2000.

[2] Das, N., Shanbhogue, M., Chen, S.T., Hohman, F., Chen, L., Kounavis, M.E. and Chau, D.H.,. "Keeping the bad guys out: Protecting and vaccinating deep learning with jpeg compression". *arXiv preprint arXiv:1705.02900*, 2017.

[3] Dziugaite, G.K., Ghahramani, Z. and Roy, D.M.,. "A study of the effect of jpg compression on adversarial images." *arXiv preprint arXiv:1608.00853*, 2016.

[4] Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M. and Thrun, S,. "Dermatologist-level classification of skin cancer with deep neural networks." *Nature*, *542*(7639), p.115, 2017.

[5] Field, D.J., Relations between the statistics of natural images and the response properties of cortical cells. *Josa a*, *4*(12), pp.2379-2394.1987.

[6] Guo, C., Rana, M., Cisse, M. and van der Maaten, L, "Countering adversarial images using input transformations." *arXiv preprint arXiv:1711.00117*, 2017.

[7] Janai, J., Güney, F., Behl, A. and Geiger, A.," Computer vision for autonomous vehicles: Problems, datasets and state-of-the-art". *arXiv preprint arXiv:1704.05519*. 2017

[8] Kurakin, A., Goodfellow, I. and Bengio, S.,"Adversarial examples in the physical world." *arXiv preprint arXiv:1607.02533*. 2016

[9] Lin, Y.C., Hong, Z.W., Liao, Y.H., Shih, M.L., Liu, M.Y. and Sun, M.,"Tactics of adversarial attack on deep reinforcement learning agents." *arXiv preprint arXiv:1703.06748*. 2017.

[10] Luo, Y., Boix, X., Roig, G., Poggio, T. and Zhao, Q.,"Foveation-based mechanisms alleviate adversarial examples." arXiv preprint arXiv:1511.06292. 2015.

[11] Moosavi-Dezfooli, S.M., Fawzi, A., Fawzi, O. and Frossard, P., “Universal adversarial perturbations.” *arXiv preprint*. 2017.

[12] Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z.B. and Swami, A, “The limitations of deep learning in adversarial settings. In *Security and Privacy” IEEE European Symposium on* (pp. 372-387). IEEE. *2016*

[13] Simoncelli, E.P.,”Bayesian denoising of visual images in the wavelet domain. In Bayesian inference in wavelet-based models” (pp. 291-308). Springer, New York, NY. 1999

[14] Song, S., Chen, Y., Cheung, N.M. and Kuo, C.C.J., Defence Against Adversarial Attacks with Saak Transform.” arXiv preprint arXiv:1808.01785. 2018.

[15] Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D. and McDaniel,P.,”Ensemble adversarial training: Attacks and defenses.” *arXiv preprint arXiv:1705.07204*. -2017.

[16] Wang, B., Lin, A.T., Shi, Z., Zhu, W., Yin, P., Bertozzi, A.L. and Osher, S.J.,“Adversarial Defence via Data Dependent Activation Function and Total Variation Minimization.” arXiv preprint arXiv:1809.08516. 2018

[17] Xie, C., Wang, J., Zhang, Z., Ren, Z. and Yuille, A.,”Mitigating adversarial effects through randomization.” arXiv preprint arXiv:1711.01991. 2017

[18] Yen. C., Hong, Z.W., Liao, Y.H., Shih, M.L., Liu, M.Y. and Sun, M,”Tactics of adversarial attack on deep reinforcement learning agents.” *arXiv preprint arXiv:1703.06748*. 2017.

[19] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A. and Torralba, A.,”Learning deep features for discriminative localization.” In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2921-2929) 2016.