**Abstract:**

The objective of this project was to classify if the image belongs to class A or class B using a machine learning algorithm.

**Theory**

1. The objective of this project is to train a machine learning algorithm to classify the images into two classes.
2. In this problem, machine learning algorithm used was SVM with polynomial kernel with degree = 2. Algorithm was chosen over Logistic regression because of SVMs non-linear approach.
3. Kernel was chosen to be polynomial with degree=2 using trial & error method. Other kernels used were RBF and polynomial with degree = 3. Accuracy was significantly hindered for these kernels.
4. Python code implemented to achieve this objective revolves around four major factors
5. **Data pre-processing:** We started by importing the images from both the folders. To reduce the complexity of the code, as soon as a single image was imported from the folder, they were normalized with mean zero and divided by standard deviation and resized to 300x300 to maintain consistency, as there were varying resolutions. 300x300 was chosen using trial & error.
6. **Data-set generation:** Imported images were saved in two arrays which were vertically stacked into X and Y was generated manually by labelling each image ‘A’ or ‘B’ depending on the folder of the images. Then, X & Y were randomly shuffled to avoid over-fitting. X was divided into 80% training data-set and 20% test data-set.
7. **Machine Learning:** The training data-set was fitted to SVM with polynomial kernel of degree two. We used sklearn library in python to achieve the same.
8. **User Input:** Diagnose function is the user input which takes in the path of the image and outputs either ‘A’ or ‘B’.

**Observation**

1. In the previous section, we can see that the images were trained 80% random shuffled data with the other 20% data to test the trained model.
2. Images are classified with more than 90% accuracy for this model for 10 iterations of testing.

**Conclusion**

1. We have achieved a successfully trained SVM model with polynomial kernel of degree 2.
2. Model is trained on 80% data set which gives us four images for testing. Accuracy was 94% when random test data with four images was passed through the algorithm.

**Future Scope:**

1. If the number of images is more, we can reduce the dimensionality by using PCA because 300x300 is comparatively a higher resolution and PCA would help reduce time and space complexity.
2. We can also train the model with key-points rather than the images themselves. SIFT algorithm in OpenCV outputs key-points and descriptors with 128 elements which significantly reduce time and space complexity. The paper to achieve this objective is referenced in the references.

**References:**

1. <https://www.sciencedirect.com/science/article/pii/S1877050915035620>