Spectacle Frame Recommendation by Detection Of Face Shape

Patel Nilay Prakashbhai 200001060
Pranjali Arun Gavhale 200001024
Sakshi Mahajan 200001041

February 22, 2023

Problem Definition: *The goal of this project is to develop a system that can automatically classify human faces into 5 different shape categories, such as round, oval, square, oblong or heart-shaped. The system will take as input a collection of facial images in .jpeg format, and use deep learning techniques to extract and analyze facial features, allowing it to accurately classify each image according to its corresponding face shape category. The project will involve building a CNN-based model to train and test the face shape classification system, as well as implementing various performance evaluation metrics to assess the effectiveness of the model. The final system will then recommend spectacles frame according to the face shape. For example, square frame looks good for round face shape.*

1 Data Collection and Preprocessing

We have collected the images and labelled them accordingly. We have 800 images for each face shape for training and 200 for testing. This makes a total of 5000 images to be trained and then tested.

Preprocessing is an essential step in developing an AI system for classifying faces according to their shape. The preprocessing step involves transforming the raw image data into a format that is suitable for further analysis and classification by the AI system. Here are some steps you can take to preprocess the .jpeg format images for your project:

1. Convert the .jpeg images to HSV: The first step in preprocessing is to convert the images to HSV. This helps to detect the change in intensities in the images.
2. Convert the HSV images to grayscale: The second step in preprocessing is to convert the images to grayscale. This helps reduce the dimensionality of the images, making it easier for the AI system to process them
3. Resize the images: To ensure that the AI system can process the images efficiently, you need to resize them to a standard size. This will help ensure that the images are consistent in terms of dimensionality and will enable the system to analyze them more effectively.
4. Normalize the pixel values: The next step is to normalize the pixel values of the images. This means scaling the pixel values so that they are in a consistent range, usually between 0 and 1. This will help ensure that the images are consistent in terms of intensity, and will make it easier for the AI system to classify them based on shape.
5. Apply any necessary filters: Depending on the specific requirements of your project, you may need to apply filters to the images to enhance certain features. For example, you may need to apply edge detection filters to highlight the edges of facial features.
6. Crop the images: Finally, you may need to crop the images to focus on the face and remove any extraneous background information. This will help ensure that the AI system is analyzing only the facial features and not any other elements in the image.

By following these steps, you can preprocess the .jpeg format images for your project and prepare them for analysis and classification using an AI system.

2 Study and Understanding of Algorithm

We would use a CNN Model for Detecting face shape Here is an overview of the algorithm for using a CNN on preprocessed images for face shape detection:

1. First, you need to preprocess the input images to make them suitable for feeding into the CNN. This can involve steps such as resizing the images to a uniform size, normalizing the pixel values, and converting the images to grayscale.
2. Next, you need to split the preprocessed images into training, validation, and test sets. The training set is used to train the CNN, the validation set is used to evaluate the performance of the CNN during training and make adjustments to the model if necessary, and the test set is used to evaluate the final performance of the CNN.
3. Define the architecture of the CNN. This involves specifying the number and type of layers in the network, the activation functions used in each layer, and the connections between layers.
4. Train the CNN using the preprocessed training set. During training, the CNN learns to recognize patterns in the images that correspond to different face shapes.
5. Evaluate the performance of the CNN using the validation set. This can involve measuring metrics such as accuracy, precision, recall, and F1 score.
6. Fine-tune the model if necessary by adjusting the hyperparameters of the CNN, such as the learning rate and the number of epochs.
7. Test the final performance of the CNN using the preprocessed test set. This can involve measuring the same performance metrics as in step 5.
8. Use the trained CNN to classify new images of faces according to their shape. This involves feeding the preprocessed image into the CNN and obtaining the predicted shape label from the output layer.

Note that there can be many variations and nuances to the above algorithm depending on the specific requirements of the face shape detection task and the nature of the preprocessed images.

The number of layers used in a CNN for face shape detection can vary depending on the specific requirements of the task and the complexity of the images. However, typically a CNN used for image classification tasks such as face shape detection would consist of multiple layers, including convolutional layers, pooling layers, and fully connected layers.

A typical CNN architecture for image classification may have several convolutional layers, followed by pooling layers to downsample the feature maps and reduce their spatial dimensions. The output of the last pooling layer is then flattened and fed into one or more fully connected layers to produce the final output.

For face shape detection specifically, it is common to use a pre-trained model as a starting point and fine-tune it on the specific task at hand. We would use a pre-trained model such as VGG16 , which have many layers and have been trained on large datasets of natural images, and then adapt the model for face shape detection by modifying the output layer and fine-tuning the weights on a dataset of preprocessed face images.

3 Study of Performance Measurement criteria

When evaluating the performance of a face shape classification model, there are several criteria you can use to measure its effectiveness. Here are some of the most common:

1. Accuracy: This is a measure of how often the model correctly predicts the face shape for a given image. It is defined as the number of correct predictions divided by the total number of predictions.
2. Precision: This is a measure of the proportion of true positive predictions (correctly classified faces) out of the total positive predictions (all predicted faces of a particular shape). It is defined as true positives divided by true positives plus false positives.

3. Recall: This is a measure of the proportion of true positive predictions out of the total number of actual faces of a particular shape. It is defined as true positives divided by true positives plus false negatives.

4. F1 score: This is a measure that combines precision and recall to give a single performance metric. It is defined as the harmonic mean of precision and recall.

5. Confusion matrix: This is a table that summarizes the number of true positives, true negatives, false positives, and false negatives for each class, and is used to visualize the performance of the model.

6. ROC curve and AUC: The receiver operating characteristic (ROC) curve is a plot of the true positive rate (sensitivity) versus the false positive rate (1-specificity) for different classification thresholds. The area under the curve (AUC) is a measure of the overall performance of the model, with a higher AUC indicating better performance.

When evaluating the performance of the face shape classification model, it's important to use multiple criteria to get a more complete picture of its effectiveness.