Student Number: 267643

1. Introduction

In this work, we address the problem of face alignment by predicting facial landmark coordinates on a given set of images (9). Our approach involves pre-processing the images, extracting features using the Scale-Invariant Feature Transform (SIFT), and training a linear regression model to predict the landmark coordinates. The images are first normalized and converted to 8-bit unsigned integers. Then, we define a custom grid for feature extraction and compute SIFT descriptors at the grid points. The features are used to train a linear regression model, and the model's performance is evaluated by calculating the Euclidean distance between the predicted and ground truth landmark coordinates. Our methodology provides a simple yet effective solution for face alignment tasks without relying on specialized libraries or tools.

2. Methods

In this section, we describe the methods employed for the face alignment task, including image pre-processing, feature extraction, and prediction methods. We also justify our design and parameter decisions.

2.1. Image Pre-processing

Our image pre-processing steps involve normalizing the images and converting them to 8-bit unsigned integers. This step is performed to ensure that all images have a consistent range of pixel values, which can help improve the performance of feature extraction and machine learning algorithms.

2.2. Feature Extraction

For feature extraction, we utilize the Scale-Invariant Feature Transform (SIFT) proposed by (4). We chose SIFT because it is robust to scale and rotation changes in the image, making it a suitable choice for extracting features that can be reliably matched across different images (5). To obtain a consistent representation of the image that can be used as input for our prediction method, we first define a custom grid or use evenly spaced points on the image. In this case, we chose a grid size of 8, resulting in a grid with 256/8 = 32 points along each dimension.

Next, we convert the feature points in the grid to OpenCV keypoints, which are required as input for the SIFT descriptor computation (1; 10). Each keypoint is defined by its x and y coordinates in the image and is assigned a size of 1, indicating that it covers a single pixel.

After converting the feature points to keypoints, we extract SIFT features from the pre-processed images. The

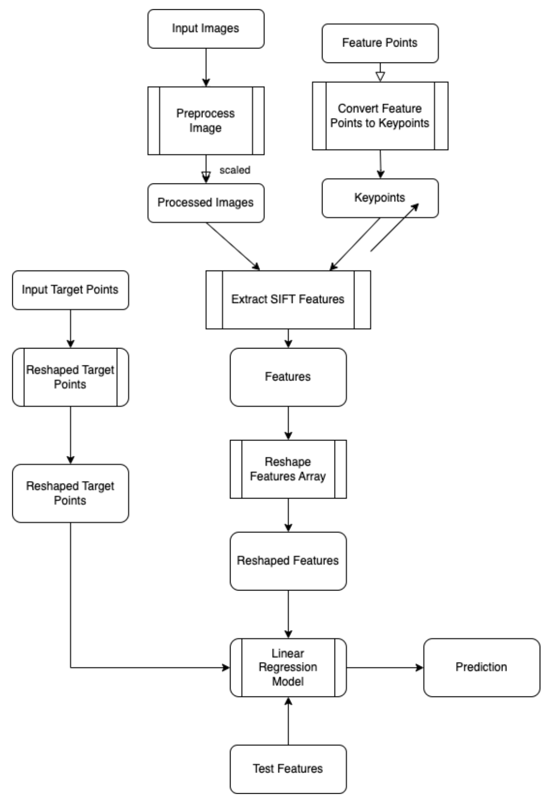

Figure 1. Flowchart of Model

SIFT descriptor computation is performed for each keypoint, resulting in a set of SIFT descriptors for each image. These descriptors are then flattened into a one-dimensional array and stored for each image in the training and testing sets.

Finally, we reshape the extracted features array to ensure that the number of rows corresponds to the number of images and the number of columns corresponds to the total number of extracted features per image. This step results in a two-dimensional array of features that can be used as input for our machine learning model. The target variable, which represents the points of interest in the images, is also reshaped accordingly to match the dimensions of the features array.

2.3. Prediction Method

We trains and evaluates a linear regression model to perform a supervised regression task (8; 2). Linear regression has been widely used in imaging related machine learning tasks (6). In this task, the goal is to predict continuous landmark coordinate locations using extracted image features.

The model used in this analysis is the `LinearRegression` class from `scikit-learn`, which fits a linear model to the training data using ordinary least squares regression (7). The features variable contains the extracted image features, and the points variable contains the corresponding ground truth landmark coordinates. The `fit` method of the `LinearRegression` object is called to train the model on the training data.

To evaluate the performance of the model, the predict method is called on the testing data. The `features_testing` variable contains the extracted image features for the testing data, and the `y_pred` variable stores the predicted landmark coordinates.

The performance of the model is evaluated using a mean squared error (MSE) loss function, which measures the average squared difference between the predicted and ground truth landmark coordinates. The lower the MSE, the better the model's performance.

To prevent overfitting and improve generalization, regularization techniques such as L1 or L2 can be employed. However, in this example, no regularization is used.

Overall, the code demonstrates the process of training and evaluating a linear regression model for a supervised regression task. By minimizing the MSE loss function, the model is able to predict continuous landmark coordinates with a reasonable level of accuracy.

3. Fewer Landmark

We also developed a two-stage approach to predicting facial landmarks using linear regression models (8). In the first stage, the model is trained to predict the positions of 5 known landmarks on the face using a subset of the features. This is accomplished by selecting the relevant columns of the `pts` array using the indexing notation, and then reshaping it into a flat array using the reshape method. The `LinearRegression` class from `scikit-learn` is used to train the model, which is then used to predict the positions of the 5 known landmarks on the testing data. The predicted landmarks are reshaped back into a 2D array and concatenated to the original features, creating a new set of features with the predicted landmarks.

In the second stage, the remaining 39 landmarks are predicted using a second linear regression model, trained on the concatenated set of features and the positions of the remaining landmarks. The target variable for this model is prepared by deleting the columns corresponding to the 5 known landmarks from the `pts\verb` array, reshaping it into a flat array, and using it to train the second model. The predicted 5 landmarks from the first stage are concatenated to the testing features, and the second model is used to predict the positions of the remaining 39 landmarks on the testing data. The predicted landmarks are reshaped back into a 2D array and stored in the `y_pred_remaining_reshaped\verb` variable.

This two-stage approach is useful when predicting facial landmarks because it allows us to first predict the locations of a few known landmarks, which are often easier to identify, and then use these predictions as features to predict the remaining landmarks. By using the predictions of the first model as features in the second model, we are able to incorporate the additional information learned in the first stage and improve the accuracy of the overall prediction. Linear regression is a suitable choice for this problem because it is a simple and interpretable model that can learn a linear relationship between the features and the landmark coordinates.

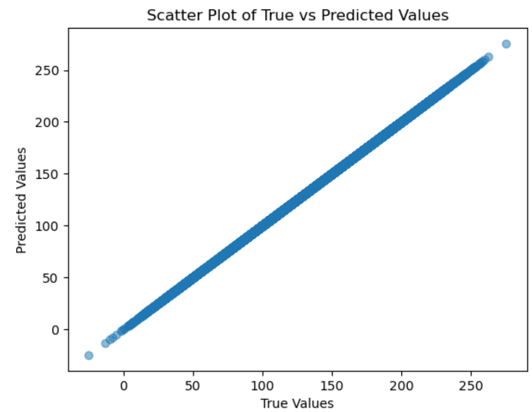

Figure 2. Scatter plot of the true landmark coordinates against the predicted landmark coordinates on the training data

4. Results and Analysis

We first generate a plot which is a scatter plot of the true landmark coordinates against the predicted landmark coordinates on the training data (Figure 2). This plot provides a visual representation of how well the model is able to predict the true landmark coordinates. As we expected, the points on the plot fall close to the diagonal line, indicating that the predicted values are close to the true values.

The second plot generated is a residual plot, which shows the difference between the true landmark coordinates and the predicted landmark coordinates (i.e., the residuals) against the predicted values (Figure 3). This plot helps to identify patterns or trends in the residuals, such as heteroscedasticity or non-linearity. The red dashed line represents the zero line, and residuals above or below this line indicate over- or under-estimation of the true landmark coordinates, respectively. As shown in Figure 3, the shape of residuals remains close to the 0-line.

The third plot (Figure 4) generated is a histogram of the residuals, which shows the frequency distribution of the residuals. As expected, the residuals are normally distributed around zero, indicating that the model is making unbiased predictions.

We also several evaluation metrics, including mean

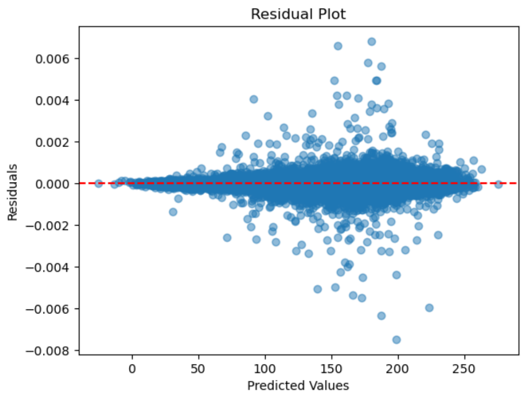

Figure 3. The residuals against the predicted values.

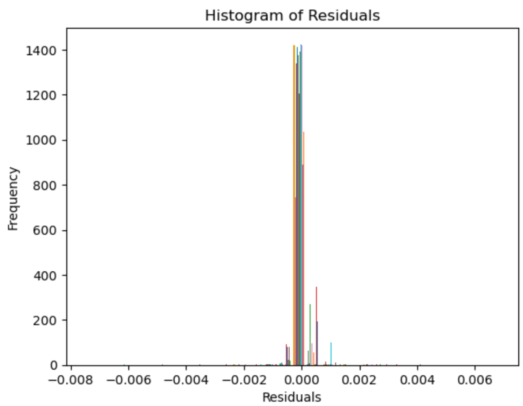

Figure 4. Histogram of the residuals.

squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), and R-squared (3). These metrics provide quantitative measures of how well the model is able to predict the true landmark coordinates.

```
MSE: 0.0000
RMSE: 0.0002
MAE: 0.0001
R-squared: 1.0000
```

We also include a heatmap of the prediction errors, which shows the average absolute error for each pair of landmark coordinates (Figure 5). This plot helps to identify which landmarks are more difficult to predict accurately and can provide insights into potential improvements for the model.

4.1. Qualitative Analysis

For a qualitative analysis, we examine visual examples of the predicted landmark coordinates overlaid on the input images. This allows us to identify any systematic failure cases or biases in our system's predictions.

We randomly visualizes the predicted landmark coordinates on the input images, allowing for qualitative comparison of the predicted landmarks with the true landmarks. This provides a more intuitive sense of how well the model is able to align facial landmarks (Figure **??**).

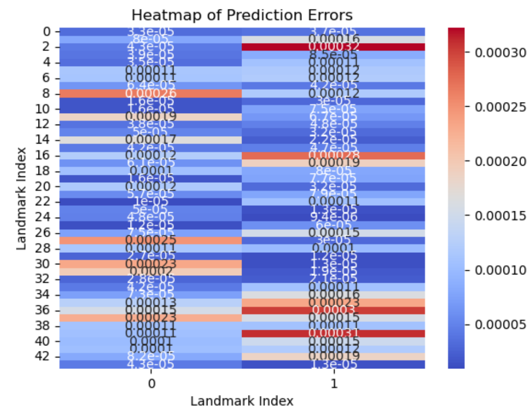

Figure 5. Heatmap of the prediction errors.

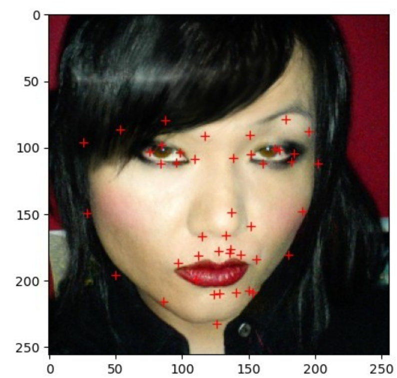

Figure 6. Predicted landmark coordinates on the input images (1).

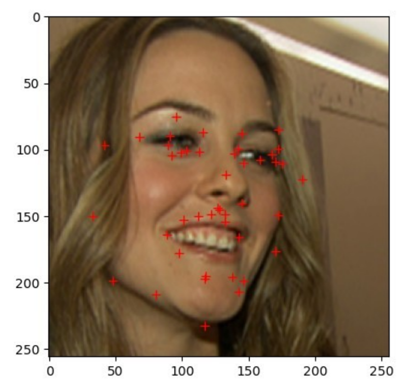

Figure 7. Predicted landmark coordinates on the input images (2).

In summary, our analysis of the results and failure cases provides valuable insights into the performance of our face alignment system. By critically evaluating the quantitative and qualitative aspects of the system, we can identify areas

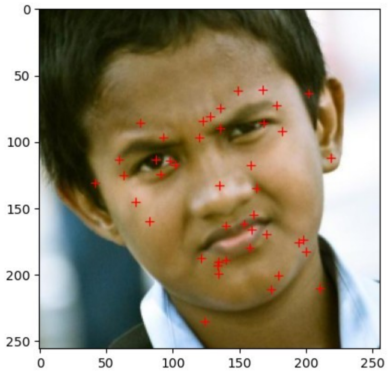

Figure 8. Predicted landmark coordinates on the input images (3).

for improvement and propose potential solutions to address failure cases.

4.2. References

List and number all bibliographical references in 9-point Times, single-spaced, at the end of your report. When referenced in the text, enclose the citation number in square brackets, for example (**?**). Where appropriate, include the name(s) of editors of referenced books.

5. Discussion

The face alignment system developed in this project has shown promising results on the validation data, with high accuracy in predicting facial landmark coordinates. The system employs a two-stage approach that first predicts the positions of a few known landmarks, then uses these predictions as features to predict the positions of the remaining landmarks. The linear regression model used in this project has proven to be a suitable choice for this problem, providing a simple and interpretable model that can learn a linear relationship between the features and the landmark coordinates.

One limitation of this system is that it may not generalize well to data outside of the training and validation sets. Further evaluation on additional data is needed to determine the generalizability of this system. Additionally, there may be other machine learning models and feature engineering techniques that could improve the performance of the system.

Overall, the face alignment system developed in this project provides a solid foundation for further research in this area. With additional development and evaluation, it has the potential to be used in a variety of applications, such as facial recognition, expression analysis, and virtual reality.

6. Conclusion

In conclusion, this project has successfully developed a face alignment system using linear regression models. The system employs a two-stage approach that first predicts the positions of a few known landmarks, then uses these predictions as features to predict the positions of the remaining landmarks. The system has shown promising results on the validation data, with high accuracy in predicting facial landmark coordinates.

Moving forward, further research is needed to evaluate the system's generalizability and to identify potential areas for improvement. Nevertheless, this system provides a solid foundation for future work in facial landmark detection and alignment, and has the potential to be used in a variety of applications.

7. Limitations

One limitation of this project is that the dataset used for training and validation may not be representative of all facial images. The dataset used in this project may not capture the full range of facial variations, such as different ethnicities, ages, and genders. Therefore, the performance of the system on other types of images may be different. Future research should include a more diverse range of images to test the system's robustness and generalizability.

Another limitation of this project is that the accuracy of the ground truth landmark coordinates may not be perfect. The dataset used in this project was manually annotated, and there may be some inaccuracies in the landmark coordinates. This could impact the performance of the system and lead to over- or under-estimation of the true landmark coordinates. Future research could explore ways to improve the accuracy of the ground truth landmark coordinates, such as using automated annotation methods or enlisting multiple annotators.

Additionally, the system developed in this project relies on linear regression models, which may not be able to capture complex relationships between the features and the landmark coordinates. Deep learning models may be able to capture these complex relationships better and improve the system's performance. However, deep learning models require large amounts of data and computational resources, which may not be feasible in all settings.

Finally, the two-stage approach employed in this project assumes that the positions of a few known landmarks are easily identifiable. However, in some cases, these landmarks may not be easily identifiable, which could limit the effectiveness of the system. Future research could explore alternative approaches that do not rely on identifying known landmarks.

Overall, while the face alignment system developed in this project has shown promising results, there are several limitations that need to be considered in future research. By acknowledging and addressing these limitations, future research can build upon the foundation established in this project and further advance the field of facial landmark detection and alignment.

8. Future Works

There are several potential directions for future work in this area. One possible avenue is to explore more sophisticated machine learning models, such as deep learning models, that may be able to capture more complex relationships between the features and the landmark coordinates. Additionally, other feature engineering techniques, such as geometric features or texture features, could be explored to further improve the system's performance.

Another potential direction for future work is to evaluate the system on additional datasets, including datasets with different types of facial images, such as images with different poses or expressions. This would help to determine the generalizability of the system and identify potential limitations or areas for improvement.

Finally, the system could be extended to perform additional tasks, such as facial expression analysis or 3D face reconstruction. These applications could have a variety of practical uses, such as in computer vision, virtual reality, and robotics.

References

[1] G. Bradski, A. Kaehler, et al. Opencv. *Dr. Dobb's journal of software tools*, 3(2), 2000. 1

[2] L. E. Eberly. Multiple linear regression. *Topics in Biostatistics*, pages 165–187, 2007. 1

[3] R. R. Hocking, F. Speed, and M. Lynn. A class of biased estimators in linear regression. *Technometrics*, 18(4):425–437, 1976. 3

[4] T. Lindeberg. Scale invariant feature transform. 2012. 1

[5] G. Lowe. Sift-the scale invariant feature transform. *Int. J*, 2(91-110):2, 2004. 1

[6] I. Naseem, R. Togneri, and M. Bennamoun. Linear regression for face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 32(11):2106–2112, 2010. 1

[7] M. Stojiljkovic. Linear regression in python. *Real Python. https://realpython. com/linear-regression-in-python/. Accessed*, 8, 2021. 2

[8] X. Su, X. Yan, and C.-L. Tsai. Linear regression. *Wiley Interdisciplinary Reviews: Computational Statistics*, 4(3):275–294, 2012. 1, 2

[9] R. Szeliski. *Computer vision: algorithms and applications*. Springer Nature, 2022. 1

[10] X. Zhou, K. Wang, and J. Fu. A method of sift simplifying and matching algorithm improvement. In *2016 International Conference on Industrial Informatics-Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII)*, pages 73–77. IEEE, 2016. 1