University of Ibadan
Journal of Science and Logics in ICT Research

A Comparative Study of Age Estimation Using Edge Detection and Regression Algorithm

[1]✉ Akinwande M. T., [2]Adeyemo A. B.

Department of Computer Science
University of Ibadan
[1]*toyinmary07@gmail.com*
[2]*sesanadeyemo@gmail.com*

Abstract

Age Prediction has received a lot of attention over the years because of its numerous applications ranging from anthropology to archaeology and even forensic science. Even though there have been many methodologies developed for this purpose, it is still a problem owing to the high variability in physiological age indicators. This study compares different combinations of Edge Detection and Regression algorithms to determine the best possible way to predict the apparent age of an individual using the Histogram of Oriented gradients to extract features from the Detected Edges. The FGNET Dataset which contains over 1000 images of people of ages ranging from 0 to 70 years old with each individual represented at least 4 different ages was used. The Edge Detection algorithms used were Canny Edge and Sobel Filter combined with the Support Vector Regression and K-Nearest Neighbour Regression Algorithms. The performance of the Canny edge detection algorithm and the Sobel filter when combined with the HOG feature extraction algorithm were compared. It was observed that the combination of the Canny Edge Detection Algorithm and the Support Vector Regression Algorithm gave the best Predictive Accuracy.

Keywords: *Image forensics, Image processing, Age Prediction, Regression, Edge Detection.*

1. INTRODUCTION

Facial age estimation can be defined as the task of automatically assigning an exact age label (or age range) to an individual facial image [2]. Age is a numerical human attribute that grows with an individual. A person naturally has two kinds of age: Apparent age, which is the age assigned to a person based on his outward appearance, and the Chronological age, also called actual age, which is the age of person calculated by subtracting the year of birth from the current year. The advent of technology however, gave rise to another category called the Estimated age, which is the age a computer assigns to an individual based on some predefined parameters [2].

For the computer to predict human age, the assumption is that the facial image of an individual gives sufficient ageing information about such individual. This assumption has been long established as a fact from previous age estimation algorithms which employed the facial image as the primary input. In humans, the accuracy of a predicted age depends on (among several other factors) the experience and exposure of the individual who is predicting the age, for instance an individual who works with a crime investigation agency might predict human ages better than a school teacher simply because of the differences in their trainings and frequent interactions and experiences.

For a machine, however, the task is somewhat more difficult as ageing is affected by several intrinsic factors (gender, race, heredity etc.) as well as extrinsic factors (weather, drugs, condition of living etc.). Also, the temporal nature of ageing and the fact that ageing patterns are individualistic also contribute to the difficulty of age estimation as these have made it difficult

Akinwande M. T. and Adeyemo A. B. (2021). A Comparative Study of Age Estimation Using Edge Detection and Regression Algorithm Importance of Computer Science in Nigeria. *University of Ibadan Journal of Science and Logics in ICT Research (UIJSLICTR),* Vol. 6 No. 2, pp. 90 - 98

to gather facial ageing dataset suitable enough for tackling the problem [10].

Age estimation can be defined as determination of a person's age or age group. Person's age can be determined in many ways, but this research is concerned with age estimation based on two-dimensional images of human subjects. Estimated age is the age defined by computer based on person's appearance. Appearance age is usually very close to the actual or chronological age. The objective of age estimation is that estimated age is as close to appearance age as possible [7].

Age estimation from face images is an interesting computer vision problem. It has various applications, ranging from customer relations to biometrics and entertainment. There have been several approaches to solving the problem of age estimation but one of the most popular approaches is a combination of edge detection and regression techniques. K-Nearest Neighbour and Support Vector regression are two of the most popular regression algorithms used in this area of research while the Sobel operator or Sobel filter has been observed to be the most popular edge detection technique [10].

2. Related Works

Over the past decade, many approaches have been proposed for age estimation. Aging pattern has been defined by Geng *et. al.* [6] as the sequence of a particular individual's face images sorted in time order, by constructing a representative subspace. A manifold learning method for age estimation was proposed by Guo *et. al.* [3]. The original face image space was mapped into a low-dimensional subspace by using special subspace learning method. Then, they designed a locally adjusted robust regression algorithm to learn and predict human age. Afterwards, Guo and Wang [5] found that there existed relations between age prediction and expression changes and also built a robust system to conduct cross-expression age estimation.

Wu *et. al.* [13] made emphasis on facial shapes, which were modelled as landmarks on Grassmann manifold. Then, these points were projected onto tangent space, and age estimation was fulfilled by Tangent Space Regression. Guo *et. al.* [4] established hybrid features, combining features, such as shape feature, texture feature, and frequency feature. They utilized SVM and SVR to predict human age. One of the first attempts to develop facial age estimation algorithms was reported by Kwon and Lobo [8]. They used two main types of features: Geometrical ratios calculated based on the distance and the size of certain facial characteristics and an estimation of the number of wrinkles detected by deformable contours (snakes) in facial areas where wrinkles are usually encountered.

Based on these features they classified faces into babies, adults and seniors. Age estimation using CNN architectures have been carried out by Yang *et. al.* [15], Levi and Hassner [9], Xing *et. al.* [14], Wan *et. al.* [11], Dong *et. al.*and Wang *et. al.* [12], with varying degrees of success. Some of the CNN architectures developed include, Shallow CNN, Deep CNN or Deep ConvNets and Multi-Task Deep Models, can also be used for gender and race determination.

Geng *et. al.* [6] generated aging patterns for each person in a dataset consisting of face images showing each subject at different ages. Each collection of temporal face images is considered as a single sample, which can then be projected to a low dimensional space. Given a previously unseen face, the face is substituted at different positions in a pattern and the position that minimizes the reconstruction error indicates the age of the subject. Experimental results based on publicly available datasets prove that this method outperformed previous approaches reported in the literature and also performed better than widely used classification methods. The results of this work suggest that methods that aim to deal with the unique characteristics of aging can yield better results when compared to standard classification approaches.

Aging is an uncontrollable personalized process which differs from one person to another. Facial age estimation techniques are extensively used in vitality applications nowadays; however, there are lots of methods of estimating human age. Age estimation is a challenging problem since face characteristics such as hair, muscles and wrinkles change over time.

3. Methodology

Figure 1 presents the schematic diagram of the methodology used for the study. A typical image processing methodology follows a process that generally involves pre-processing, which, in most case involves noise removal and possibly resizing images to the same size for consistency. This is then followed by a feature extraction stage for which there are several methods of extracting features from an image, some of which are a combination of algorithms, this research employs a combination of two techniques which are edge detection and histogram of oriented gradients (HOG).

The performance of the Canny edge detection algorithm and the Sobel filter was compared when combined with the HOG feature extraction algorithm. Age is a continuous variable and as such the problem of age prediction is a regression problem. This means that the goal is to predict numerical outcomes based on the features generated from the images and to compare the performance of the different combinations of edge detection and regression algorithms for age prediction.

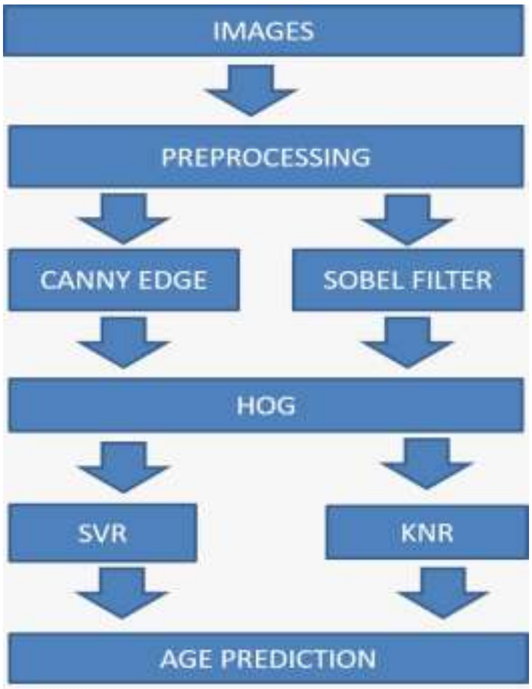

Figure 1: Schematic diagram for the methodology

The dataset used in the study comprised of over 1000 images of people of ages ranging from 0 to 70 years old. The images are a subset of the FG-NET Aging database. Each image is stored in jpeg format with the age of the person as the last 2 characters of the name of the name of the file. For example, if the file name is "001A40" it means that the person in the image is 40 years old. This forms the labels for the dataset. Figure 2 shows an example of the images in the dataset. As a pre-processing step, the images are converted to grayscale and resized so that they are all the same size for the sake of consistency.

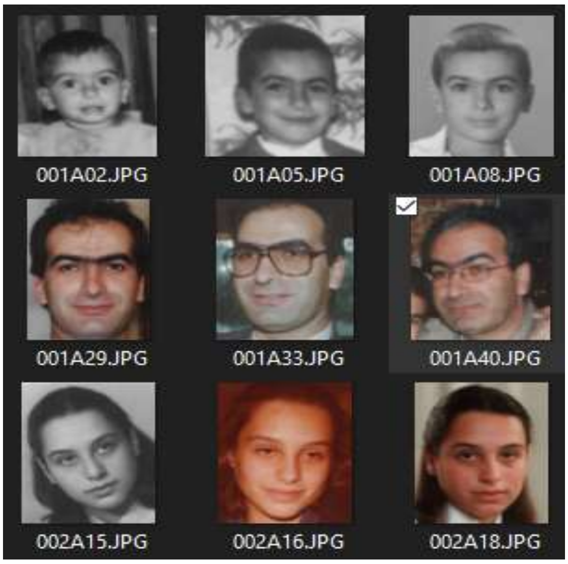

Figure 2: Sample of Images in Dataset

3.1 Feature Extraction

Edge detection algorithms were used in the study to extract features from the images in the dataset. Two edge detection algorithms are observed in this research. Before using these algorithms on the image, each image is passed through a gaussian filter to smoothen out the images and remove noise because edge detection algorithms are notorious for being affected by noise in images.

3.2 Canny Edge Detection

The canny edge detector is a multistage algorithm that is used for detecting a wide range of edges on an image. It involves the following steps

- Pre-processing: Generally, edge detectors are prone to noise and as such require a noise filter in order to operate optimally. In this research, a gaussian filter is applied to the images to reduce the noise that would otherwise affect the performance of the canny edge detector.

- Gradient Calculation: gradient magnitudes and directions are then calculated at every single point in the image. The magnitude of the gradient at a point determines whether or not it lies on an edge or not. If the gradient magnitude is high that there is a rapid change in color implying and edge whereas reverse is the case for a low gradient magnitude.

The magnitude is calculated for both the horizontal and vertical axis using a simple sobel operator. Gradient direction is always perpendicular to edges. It is rounded to one of four angles representing vertical, horizontal and two diagonal directions.

- Non-Maximum Suppression: after the gradients have been calculated, the whole image is the scanned and pixels which may not constitute edges are then removed. To do this, every pixel is checked to see if it is the local-maxima in the neighborhood of the direction of the calculated gradient. These pixels are then suppressed as they most likely do not form part of an edge in the image.

- Hysteresis Threshold: Even after the Non-Maximum Suppression stage, there may still be some false edges. To remedy this, a threshold value is set such that only gradient magnitudes that are above this threshold are considered edges.

However, is a pixel with a gradient magnitude below the threshold is connected to on above the threshold then it is also considered as part of the edge as shown in Figure 3.

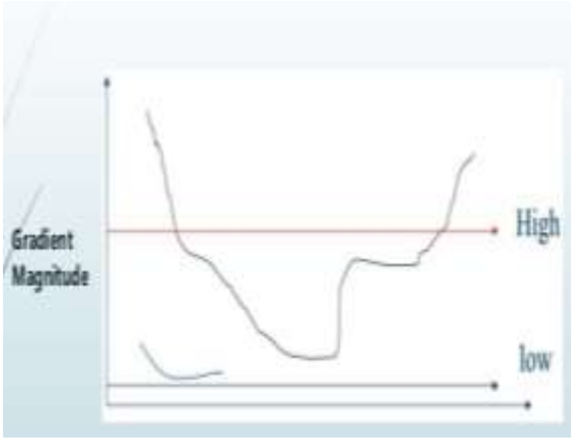

Figure 3: Hysteresis Threshold

3.3 Sobel Edge Detection

Unlike the canny edge detector, the Sobel edge detector calculates the changes in the pixels of an image. It uses a derivative mask to calculate both the horizontal and vertical edges by applying the mask to both edges and then combining the results. The mask is a typically 3 x 3 matrix consisting of differently (or symmetrically) weighted indexes. The image is then scanned, using the mask, across the X and Y axis for large changes in the gradient. A change in the gradient would imply an edge. The process is called kernel convolution and the mask can also be referred to as a kernel. The final step is combining the kernels from both axes together to form the final image showing all the edges (Figure 4).

Figure 4: Sobel Filter

After edge detection, features are now extracted using the histogram of oriented gradients (HOG). HOG is a feature descriptor that is often used to extract features from image data. It is widely used in computer vision tasks for object detection.

- The HOG descriptor focuses on the structure or the shape of an object. In the case of edge features, we only identify if the pixel is an edge or not. HOG is able to provide the edge direction as well. This is done by extracting the gradient and orientation (or you can say magnitude and direction) of the edges
- Additionally, these orientations are calculated in 'localized' portions. This means that the complete image is broken down into smaller regions and for each region, the gradients and orientation are calculated. We will discuss this in much more detail in the upcoming sections
- Finally, the HOG would generate a Histogram for each of these regions separately. The histograms are created using the gradients and orientations of the pixel values, hence the name 'Histogram of Oriented Gradients'

To calculate the HOG, first we compute the gradient vector of every pixel, as well as its magnitude and direction. Then we divide the image into many 8x8 pixel blocks. In each cell, the magnitude values of these 64 cells are binned and cumulatively added into 9 buckets of unsigned direction (no sign, so 0-180 degree rather than 0-360 degree; this is a practical choice based on empirical experiments). For better robustness, if the direction of the gradient vector of a pixel lies between two buckets, its magnitude is not all assigned to the closer one but proportionally split between the two. For example, if a pixel's gradient vector has magnitude 8 and degree 15, it is between two buckets for degree 0 and 20 and 2 is assigned to bucket 0 and 6 to bucket 20. This interesting configuration makes the histogram much more stable when small distortion is applied to the image.

Figure 5 shows how we split one gradient vector's magnitude if its degree is between two degrees bins. Finally, a 2 x 2 cells (thus 16x16 pixels) block is slided across the image. In each block region, 4 histograms of 4 cells are concatenated into one-dimensional vector of 36 values and then normalized to have a unit weight. The final HOG feature vector is the concatenation of all the block vectors. Using this feature vector we can now perform our age estimation. Figure 6 shows the resulting feature descriptors after applying the HOG to both the edges gotten from both edge detection algorithms.

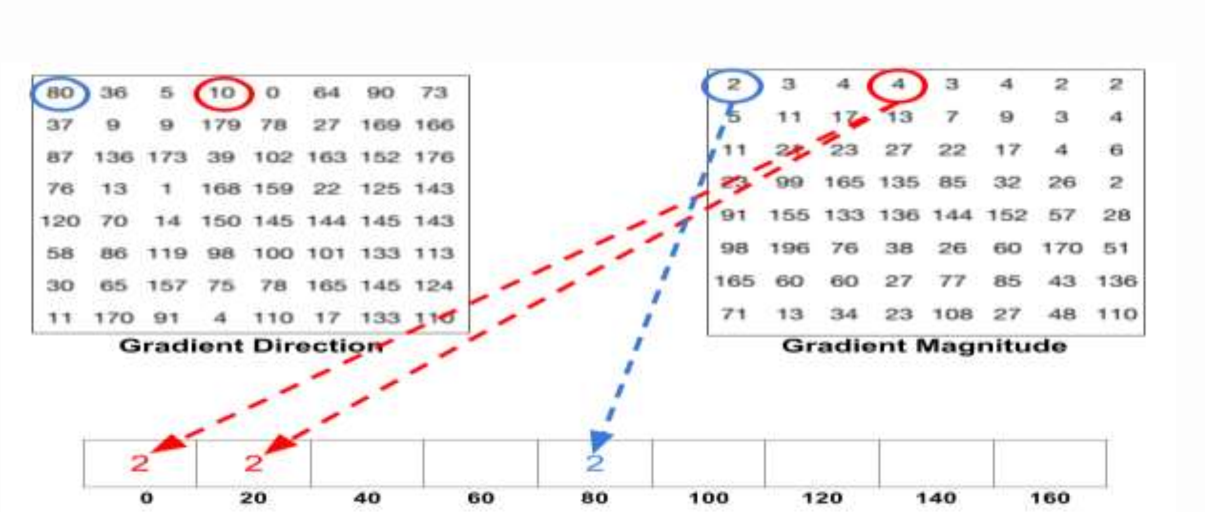

Figure 5: Histogram of Oriented Gradients (Image source: https://www.learnopencv.com/histogram-of-oriented-gradients/)

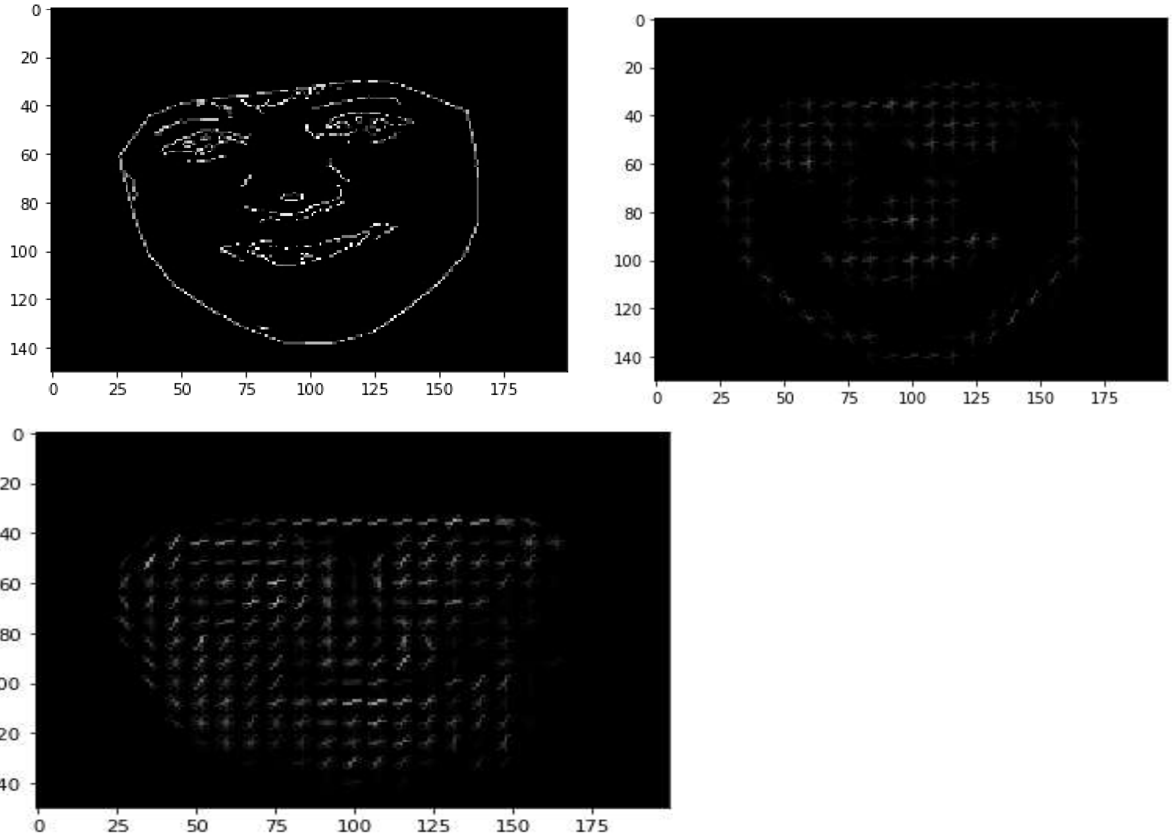

Figure 6: Edges and HOG features for Canny Edge Detection and Sobel Filter

3.4 Age Prediction

Since age is a continuous variable, the problem of age prediction is a regression problem such that the age of a person is predicted based on some features. These features have been extracted as explained previously, now to make these predictions, research has shown that the Support Vector Regression and K Nearest Neighbors algorithms are two of the most popular approaches to age prediction and as such would also be employed in this research in order to determine which combination would yield the best results.

3.5 Support Vector Regression (SVR)

The SVR is an adaptation of the Support Vector Machine Algorithm (typically used for classification) that is used to predict continuous variables. In simple linear regression, error rate is minimized while in. SVR, the error is fit within a certain threshold. First, a hyperplane is created which in SVM is a line that separates the data classes but, in this case, it will be used to predict the continuous values or target values. Next, the boundaries are then placed at +e and -e distances from the hyperplane on the Y axis as shown in Figure 7.

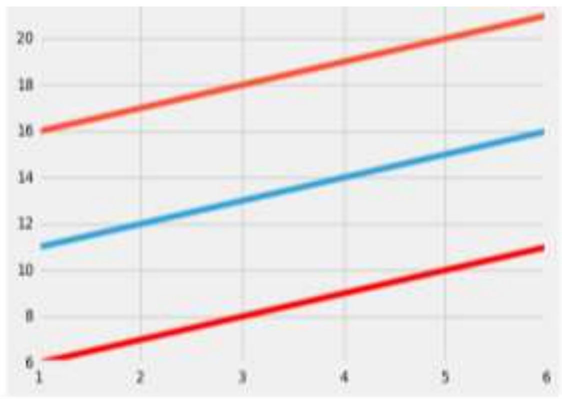

Figure 6: Support Vector Regression Hyper Plane

3.6 Nearest Neighbours Regression (KNR)

K nearest neighbours is a simple algorithm that stores all available cases and predict the numerical target based on a similarity measure (e.g., distance functions). KNN has been used in statistical estimation and pattern recognition already in the beginning of 1970's as a non-parametric technique. KNN can be used for both classification and regression problems. The algorithm uses *'**feature similarity**'* to predict values of any new data points. This means that the new point is assigned a value based on how closely it resembles the points in the training set.

- First, the Euclidean distance between the new point and each training point is calculated.
- Then we select the K closets data points based on the calculated distance
- The average of these datapoints become the predicted value of our target data point.

4. RESULTS AND DISCUSSION

The study was implemented using python programming language installed with the anaconda distribution. It used a number of python libraries some of which include: OpenCV Library – For reading images into python, Scikit-Image Library – For denoising images using a Gaussian Filter, Pandas Library – For visualizing the resulting feature vector, Scikit-Learn Library – For performing both SVR and KNN-R algorithms, and Matplot Library – For visualizing both our images and the results of the experiments. Jupyter notebook which is a part of the anaconda distribution was also used to run scripts in this research which allows for the python scripts to be run piece by piece.

This enables easier debugging and incremental development. Four different experiments were carried out, alternating between edge detection algorithms and regression algorithms. The canny edge detection algorithm and its resulting feature vector based on HOG are fed in to both the support vector regression algorithm and the K-Nearest Neighbors Regression algorithm. This was repeated using the edges detected by the sobel filter. The metrics used for the evaluation of the performance of the algorithm results were the Root Mean Squared Error, Mean Absolute Error and Mean Squared Error are the metrics used to evaluate the Selecting K Value. To select the optimal k value, the algorithm was run using random values of k and the value with the lowest error rate is selected.

To carry out predictions using Canny Edge Features, after obtaining the feature vector, using HOG, from the canny edges, the features were then passed into the regression algorithms. The results shown in Figure 8 show that both regression algorithms have performances that were at par with each other.

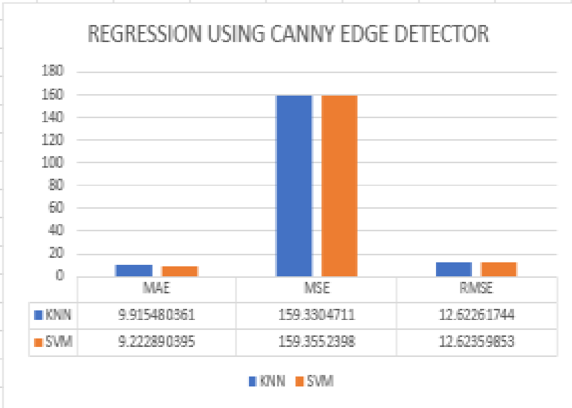

Figure 8: Regression using Canny Edge Features

To carry out predictions using Sobel Edge Features, the same regression was carried out using the features extracted from the Sobel edges. It was observed from Figure 9 that in the case of the Sobel edge features, the SVR algorithms performs slightly better than the KNN-R because it produced a lesser error value.

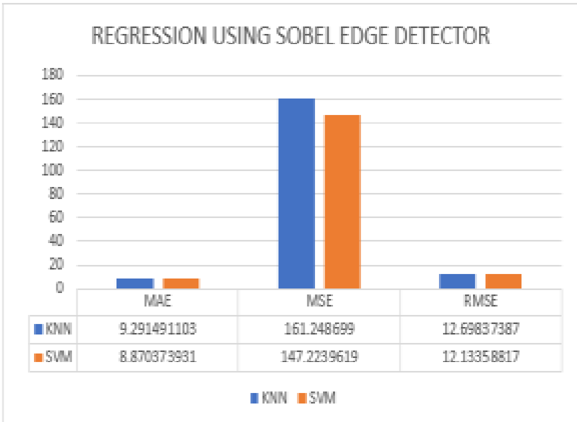

Figure 9: Regression using Sobel Features

A comparison of the Error rates for each of the four combinations carried out in this study was carried out. From Figure 10, 11 and 12 it can be observed that there is a clear distinction in the performance of the SVM/SOBEL combination from the others as it has a much lower error values than the rest of them.

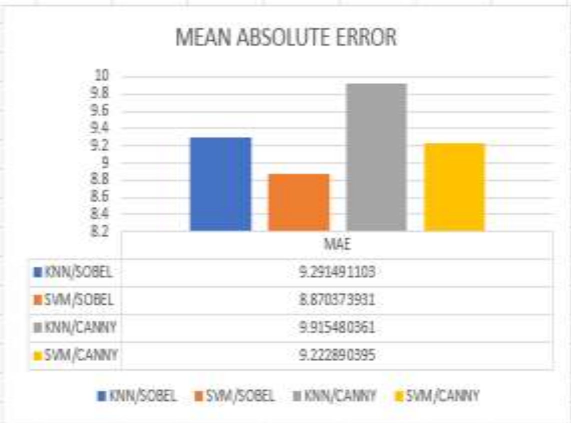

Figure 10: Mean Absolute Error

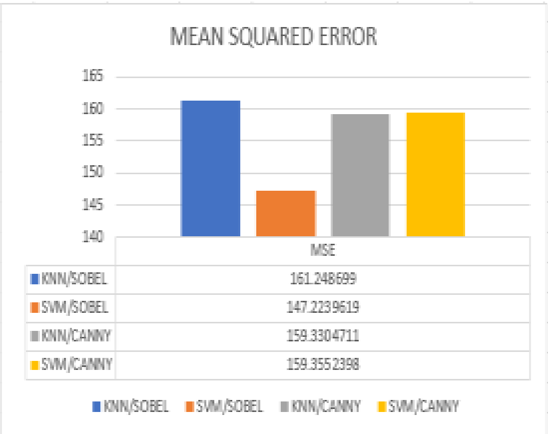

Figure 11: Mean Squared Error

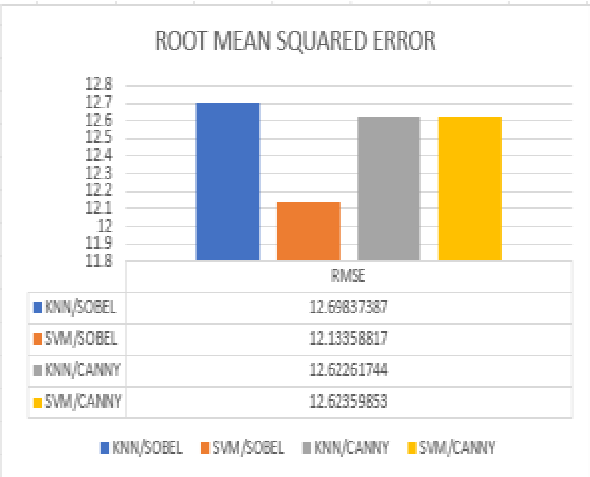

Figure 12: Root Mean Squared Error

Although the SVM/CANNY combination seems to be the next best based on the mean absolute error, the difference becomes insignificant when the errors are magnified by squaring them as seen in Figure 11.

5. Conclusion

In this study, a comparative analysis of age prediction techniques was carried out. The study investigated which combination of edge detection and regression algorithms would give the best results in predicting the age of a person from their images. Canny Edge detection and Sobel Filter algorithms were used to detect the edges in the images and the features were then extracted using the histogram of oriented gradients. These features were then passed into regression algorithms (Support Vector Regression and K-Nearest Neighbour Regression). From observed results, it can be concluded that using the Sobel Filter together with the Support Vector Regression algorithm would give a better result than the other three combinations.

References

[1] Dong Y., Liu Y., and Lian S., (2016), Automatic age estimation based on deep learning algorithm, Neurocomputing, vol. 187, pp. 4_10

[2] Fu, Y., Guo, G., analysis, T. H.-I. transactions on pattern, & 2010, undefined. (n.d.). Age synthesis and estimation via faces: A survey. Ieeexplore.Ieee.Org. Retrieved from https://ieeexplore.ieee.org/abstract/document/5406526/

[3] Guo G., Fu Y., C. Dyer R., and Huang T. S., (2008), Image-based human age estimation by manifold learning and locally adjusted robust regression. IEEE Transactions on Image Processing, vol. 17, no. 7, pp. 1178–1188

[4] Guo J .M., Liou Y. M. and Nguyen H. S., (2011), Human face age estimation with adaptive hybrid features in Proceedings of the International Conference on System Science and Engineering (ICSSE ’11), pp. 55–58, Macau, China.

[5] Guo G. and Wang X., (2012). A study on human age estimation under facial expression changes in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR ’12), pp. 2547–2553, 2012.

[6] Geng X., Zhou Z. H, and Smith-Miles K., (2007), Automatic age estimation based on facial aging patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 12, pp. 2234–2240.

[7] Jain, S., & Patil, A. J. (2016). Human Age Prediction from Facial Images, 14313–14320. https://doi.org/10.15680/IJIRCCE.2016.0407165

[8] Kwon YH, Lobo N da Vitoria, (1999), Age classification from facial images, Computer vision and image understanding, Vol 74, Issue 1, pp 1-21

[9] Levi G., Hassner T., (2015), Age and gender classification using convolutional neural networks, Proc. CVPR, vol. 24, no. 3, pp. 2622_2629. https://doi.org/10.1109/CVPR.2016.532

[10] Onifade, O. F. W. (2015). A Review on the Suitability of Machine Learning Approaches to Facial Age Estimation. International Journal of Modern Education and Computer Science, 7(12), 17–28. https://doi.org/10.5815/ijmecs.2015.12.03

[11] Wan J., Tan Z., Lei Z., Guo G., and Li S. Z., (2018), Auxiliary demographic information assisted age estimation with cascaded structure IEEE Trans. Cybern., vol. 48, no. 9, pp. 2531_2541.

[12] Wang X., Guo R., Kambhamettu C., (2015), Deeply-learned feature for age estimation, in Proc. IEEE Winter Conf. Appl. Comput. Vis. Deeply- Learned, pp. 534_541.

[13] Wu T., Turaga P., and Chellappa R., (2012), Age estimation and face verification across aging using landmarks. IEEE Transactions on Information Forensics and Security, vol. 7, no. 6, pp. 1780– 1788.

[14] Xing J., Li K., Hu W., Yuan C., and Ling H., (2017), Diagnosing Deep Learning Models for High Accuracy Age Estimation from a Single Image," Pattern Recognit., vol. 66, pp. 106_116.

[15] Yang M., Zhu S., Lv F., and Yu K., (2011), Correspondence Driven Adaptation for Human Profile Recognition," in Proc. CVPR, pp. 505_512.