Sobel Edge Detection using Optimized Image Preprocessing

Qin Tong Bettina Tee
IES24303
Nanyang Technological University
BTEE003@e.ntu.edu.sg

Daso Jung
2071043
Ewha Womans University
ekth216@ewhain.net

Yan Renyu
IES24364
Nanyang Technological University
ryan022@e.ntu.edu.sg

Abstract

This report presents a method for edge detection in a set of images. The Sobel operator is employed as the edge detection method, leveraging on image preprocessing techniques, namely Gaussian blurring and contrast enhancement, to improve its performance. Key features such as edge density, sharpness, and entropy are extracted from the image and optimal preprocessing parameters are obtained using Bayesian optimization. The extracted images features and target optimal preprocessing parameters are used to train a Random Forest Classifier. The trained predictor is used to predict the optimal preprocessing parameters and Sobel Operator and normalisation is applied to produce the resultant predicted edges. Results obtained show that this approach enhances Sobel edge detection, producing more accurate and consistent edge maps across various image types.

1. Introduction

1.1. Background and Problem Statement

Boundary detection is a fundamental task in computer vision, serving as a critical preprocessing step for various applications. However, boundary detection algorithms often struggle to adapt to the unique characteristics of different images.

2. Methodology

The goal of this project is to develop a model to predict the edges of a given image as accurately close to the given boundaries as possible. We chose to use the Sobel operator for the edge detection step. In order to enhance the performance of the edge detection, we constructed our model consisting of several stages.

2.1. Overview

This approach involves applying a series of preprocessing steps to an image before performing edge detection. These steps are designed to enhance the performance of the Sobel operator. After preprocessing the image, the sobel operator is then applied to produce the predicted edge map.

The challenge is that different images present unique characteristics which require different preprocessing parameters. We aim to predict these optimal parameters and produce the most accurate edge maps by taking the steps detailed in the next sections.

2.2. Feature Extraction

This step aims to extract essential features that can aid in the prediction of the optimal parameters that will best preprocess the image to obtain the most accurate predicted edges. The following features were hence chosen:

1. Edge Density: measures the complexity of edges in the image, using Canny edge detector.
2. Edge Sharpness: measures the clarity of boundaries in the image, by computing the average gradient magnitude using Sobel filters.
3. Colour Contrast: measures the variation in brightness levels, by computing the standard deviation of pixel intensities in the grayscale raw image.
4. Root Mean Square Contrast: measures the overall variation in intensity relative to the mean, by computing the root mean square of pixel intensities normalised by the mean intensity.
5. Entropy: measures the complexity or detail in the image, using Shannon entropy on the grayscale version of the raw image.
6. Mean Intensity: measures the image's brightness, by computing the mean of pixel values in the grayscale raw image.
7. Standard Deviation of Intensity: measures the variation of pixel intensities, by computing the standard deviation of pixel values in the grayscale raw image.
8. Colour Channel Variability: measures the consistency of variability across colour channels in the image, by computing the variance of pixel values in each colour channel.
9. Spatial Frequency: measures the level of detail or texture in the image, using Fourier Transform on the grayscale raw image.

The above features would be fed as the feature variables in our parameter prediction model.

2.3. Target Extraction

To train our model, we need to extract the target optimal parameters. We search for the best set of parameters in the defined search space using the Bayesian Optimisation algorithm, which works by building a probabilistic model of the objective function and iteratively selecting parameter sets that maximise the performance of the edge detection process. The optimal parameters found are then used as the target variables in our parameter prediction model.

2.4. Feature Importance

Having too many features in the model may cause the model to be overly complex and possibly lead to poorer performance or requiring long computational time. Hence, we use Random Forest Classifier to compute the feature importance and select the most relevant features to construct our predictive model. We obtained the following feature importance ranking:

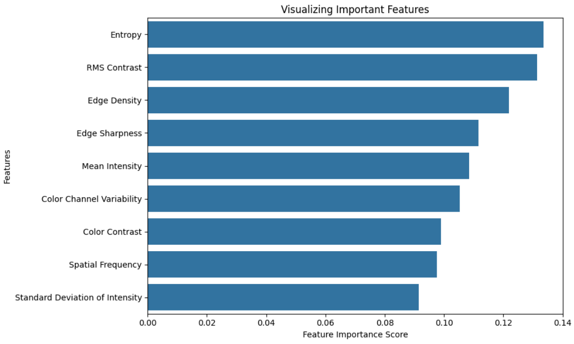

Figure 1: Feature Importance Score Graph

In order to select the optimal number of features to be used in the construction of the model, we evaluated each model with a varied number of top features from 1 through 9. The following results of accuracy as well as time taken for scoring is presented below:

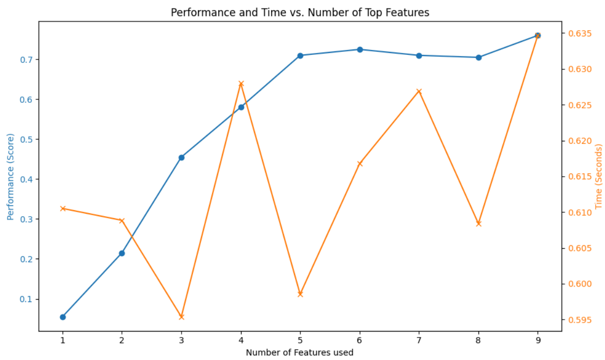

Figure 2: Accuracy and Time against number of Features

We see that the variation of the absolute values of time taken are small, hence we chose to disregard the significance of the time taken. Hence, we chose to select the number of features based solely on which gives the highest score, which is 9. In other words, we will not be reducing the number of features.

2.5. Parameter Prediction

We employ a random forest classification model to predict the optimal parameters based on the features of the image. The parameters of the random forest classification model are tuned using GridSearch. Following which, we train and fit the model using a set of images. The model will take in an image, extract the features of the image, then return a predicted set of optimal parameters, which are used in the next section.

2.6. Image Preprocessing

We first apply a set of preprocessing steps to the Image to enhance the results of the Sobel Operator outlined as follows:

1. Gaussian Blurring: reduce noise by smoothing the image with a Gaussian kernel.
    - blur_amount: controls the size of the kernel. A larger value leads to more blurring and less fine details, while a smaller value retains more detail but less smoothing.
    - sigma: standard deviation of kernel. A larger value leads to a stronger blur, while a smaller value leads to less blurring.
2. Contrast Enhancing: This step helps to adjust the image's contrast by stretching the range of pixel intensities. Increasing contrast makes features in the image more obvious, which helps the edge detection step.
    - contrast_factor: contrast enhancement factor. This parameter controls the strength of the contrast adjustment. A value greater than 1 increases the contrast while a value less than 1 reduces the contrast.

In addition, we apply the above steps to each of the colour channels separately before combining the results at the end. This the model to extract edge details specific to each colour channel, preserving colour specific edge information and colour relationships.

This results in 3 unique sets of 3 parameters, leading to a total of 9 parameters to be optimised. By completing steps detailed in section 2.2 to 2.5, we are able to obtain the optimal parameters for the above detailed preprocessing steps.

2.7. Edge Detection

2.7.1 Choice of Edge Detector

In order to select the most appropriate edge operator to be used for our set of images, we conducted experiments using 3 edge operators — Sobel, Laplacian and Prewitt. The results based on the same set of 10 images with no preprocessing is shown below:

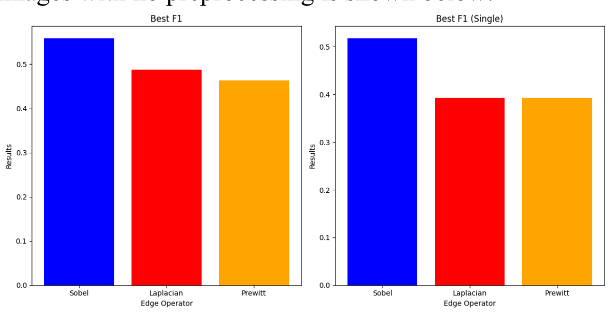

Figure 3: F1 Score Analysis of Edge Detection Operators

The graph compares the performance of the 3 edge operators based on their F1 scores in 2 scenarios: best threshold for each image and single best threshold for all images. We can see that Sobel operator achieves the best overall performance, indicating that it is the most appropriate edge operator for our set of images

2.7.2 Applying Sobel Operator

After applying the optimised preprocessing steps, we are able to apply the Sobel Operator to each colour channel to obtain their respective gradient magnitudes, which are then normalised such that the values are between 0 and 1. The edges from the 3 channels are then combined to produce the final predicted edges. The results of our model are detailed in section 3.

3. Results and Discussion

3.1. Key metrics

The performance of the edge prediction model was evaluated using three key metrics: F1-score, recall, and precision. These metrics were computed at different threshold values applied to the predicted edge maps, with the goal of determining the optimal threshold that maximizes model performance.

F1-score is the harmonic mean of precision and recall. This metric highlights the balance between false positives and false negatives. To compute the final score, two F1-scores were averaged: 1. Best F1: The F1-score calculated using the optimal threshold for each individual image. 2. Best F1 (Single): The F1-score calculated using a single optimal threshold applied to all images.

Recall reflects the proportion of actual edges correctly identified by the model. Precision measures the proportion of correctly predicted edges among all pixels classified as edges.

3.2. Comparison to without Preprocessing Steps

On the basis of a set of 10 images, the sobel operator without any prior preprocessing steps attained an average F1-score of 0.54, comprising a Best F1 of 0.56 and a Best F1 (Single) of 0.52. With our implemented training and preprocessing steps, the sobel operator attained an average F1-score of 0.64, comprising a Best F1 of 0.65 and a Best F1 (Single) of 0.63.

The preprocessing steps have led to a 16.7% increase in average F1-score, from 0.54 to 0.64. It shows that the addition of the preprocessing steps have led to a significant increase in the performance of the Sobel operator.

3.3. Results of Training Images

The evaluation on the full training set achieved an average F1-score of 0.57, comprising a Best F1 of 0.59 and a Best F1 (Single) of 0.56. For the overall evaluation metrics with the best threshold applied to each image, recall was 0.67, and precision was 0.52. Recall varied between 0.4 and 0.95 across images, while precision ranged from 0.2 to 0.9.

3.4. Training Set Analysis

The training set's average F1-score of 0.57 reflects a moderate balance between precision and recall. With a slightly higher Best F1 (0.59) compared to Best F1 (Single) (0.56), the results suggest that using an optimized threshold tailored for each image provides a marginal advantage in performance.

The recall for the training set (0.67) indicates that the model correctly identified the majority of relevant instances, although the relatively lower precision (0.52) suggests a higher proportion of false positives. The range of recall (0.4 to 0.95) and precision (0.2 to 0.9) across individual images indicates variability in performance, likely influenced by the complexity or characteristics of specific images in the dataset.

3.5. Results of Test Images

The test set submitted previously achieved an average F1-score of 0.54, comprising a Best F1 of 0.55 and a Best F1 (Single) of 0.52.

3.6. Test Set Analysis

The test set's average F1-score of 0.54 demonstrates a slight performance drop compared to the training set, as expected due to the generalization challenge. Similar to the training set, the Best F1 (0.55) surpasses the Best F1 (Single) (0.52), reinforcing the observation that image-specific thresholds enhance performance.

3.7. Comparison and Observations

The gap between the training and test F1-scores (0.03) suggests some degree of overfitting, as the model performs better on the data it was trained on. The decline in F1-score for the test set underscores the need for further refinement, such as improving feature extraction, threshold optimization methods, or augmenting the dataset for better generalization.

4. Conclusion

This report presents a method for improving edge detection in images by leveraging feature-based predictive modeling to optimize preprocessing steps for the Sobel operator. The approach effectively combines preprocessing techniques such as Gaussian blurring, and contrast enhancement with a parameter prediction model that tailors thresholds to individual image characteristics.

The results demonstrate that the proposed methodology enhances edge detection performance, as reflected in the evaluation metrics. The training set achieved an average F1-score of 0.6, with recall and precision scores of 0.67 and 0.5, respectively. The test set yielded an average F1-score of 0.54, highlighting a slight performance drop due to generalization challenges.

4.1. Strengths of the Method

The method's strength lies in its optimized preprocessing parameters, which adapt to the unique characteristics of each image to produce more accurate and consistent edge maps. By leveraging key features such as edge sharpness, edge density, and entropy, the model effectively predicts optimal parameters, significantly enhancing edge detection performance. Additionally, preprocessing individual color channels preserves fine details, leading to improved Sobel operator results and overall edge detection quality.

4.2. Limitations and Challenges

The method faces several challenges. Variability across images is a key issue, with performance metrics like recall (0.4 to 0.95) and precision (0.2 to 0.9) showing significant differences, indicating sensitivity to image characteristics. Overfitting is evident from the gap between training and test F1-scores, suggesting the need for improved model generalization. Additionally, the approach's dependence on Bayesian optimization and multiple preprocessing steps adds computational overhead, which can be costly and limit scalability. To address these limitations, strategies must be implemented to improve model robustness, optimize preprocessing, and reduce computational costs, ensuring more consistent, reliable, and efficient performance across varied image data.

4.3. Recommendations for Future Work

Future work should refine feature selection to improve parameter prediction, explore data augmentation to enhance model generalization, and adopt adaptive threshold optimization for better precision-recall balance. Additionally, improving computational efficiency through techniques like model pruning or parallel processing could accelerate predictions and preprocessing without sacrificing accuracy. These efforts would enhance the robustness, accuracy, and scalability of the edge detection methodology.

4.4. Final Words

The proposed methodology demonstrates promising results in enhancing the Sobel operator's edge detection capabilities through feature-based preprocessing optimization. While there is room for improvement in generalization and efficiency, the approach offers a robust foundation for further exploration in edge detection and related computer vision tasks.