Signal and Image Processing - Mod. 1

Report for the July 2024 exam

Project members

Patrizio Acquadro	502311	patrizio.acquadro01@universitadipavia.it
Lorenzo Uttini	502183	lorenzo.uttini01@universitadipavia.it

1 Introduction

The Image Processing project involves completing two distinct tasks: classification and image correction. We began by collecting a dataset of 138 images, categorized into three classes representing different levels of exposure (normal exposure, underexposure, and overexposure). For the classification task, we extracted various features from the images and used these to train several models. In the image correction task, we experimented with three different methods to determine the best gamma value for correcting improperly exposed images. We then evaluated the gamma-corrected images using the most effective classifier identified during the classification task. Additionally, we developed a new method based on entropy to assess the performance of the gamma correction.

2 Dataset

The dataset consists of **138 images** captured using an iPhone 12, which features a 12 MP main camera and a 12 MP ultra-wide camera. These images were taken in various environments, both indoor and outdoor, and under diverse lighting conditions, including day, night, natural light, artificial light, and rainy conditions. The subjects in the images include people, objects, and animals, with some images forming triplets that show the same subject under correctly exposed, overexposed, and underexposed conditions. This diversity ensures that the classifier developed is robust and based on exposure rather than specific subject patterns, leading to more reliable results in different scenarios.

The dataset is perfectly *balanced with 46 images* in each of the three classes: correctly exposed, underexposed, and overexposed. This equal distribution is crucial for unbiased training and evaluation in subsequent image processing and classification tasks, allowing for a fair comparison between the different exposure categories.

For the purpose of image correction, the images underwent pre-processing where they were converted to double precision and transformed into the YCbCr color space. The use of double precision normalizes pixel values within the range of 0 to 1, essential for gamma correction's functioning. Transforming the images into the YCbCr color space separates luminance (Y) from chrominance (Cb and Cr), allowing gamma correction to be applied specifically to the Y channel. This approach adjusts brightness without affecting color information. The pre-processing step is essential for maintaining the integrity of the colors while accurately correcting the luminance across the entire dataset, thus providing a solid foundation for subsequent analysis and classification. By ensuring consistency in brightness adjustments, the overall quality and effectiveness of the gamma correction process are significantly enhanced.

3 Image Classification

Classification in the image domain refers to the process of assigning a label to an image from a predefined set of categories. In this context, the categories are the exposure levels: correctly exposed, underexposed, and overexposed. In order to complete this part, we firstly extracted several features about our images, and then we trained different classifiers to pick the best one.

The extraction of features is one of the most crucial passages of our project because the extracted values will be used to train the models.

We started by extracting the first three moments for each image:

- **Mean** - Represents the average pixel intensity.
- **Variance** - Measures the spread of pixel intensities around the mean.
- **Skewness** - Indicates the asymmetry of the intensity distribution.

We apply these moments to each channel in three different color spaces:

- **RGB** - Uses Red, Green, and Blue channels.
- **HSV** - Uses Hue, Saturation, and Value components.
- **YCbCr** - Separates luminance (Y) from chrominance (Cb and Cr).

Each moment and color space provides distinct insights into the image's characteristics, aiding in exposure recognition. For example, a higher mean in RGB channels suggests a brighter, possibly overexposed image, while a lower mean suggests underexposure. High variance typically indicates a well-exposed image with good contrast. Similarly, in the HSV color space, a high mean value in the V channel suggests a bright image. This approach yields 27 different features.

Then, we created the **Pixel Intensity Histograms** for each channel of RGB images, extracting 15 features represented by five bins for each channel. Additionally, we generated 24 more features using **Edge Direction Histograms**, which were obtained by computing the gradient magnitudes and directions for each channel.

As last set of features, we decided to use the **Entropy** for each RGB channel, a measure that quantifies the amount of information or details present in the image.

Totally, we collected 69 features.

Before training the classifiers, we divided the data into training and test sets. The training set, consisting of 123 images (85% of the dataset), was used to train the classifiers, while the test set (5 instances per class) was used to evaluate the best classifier.

We trained several classifiers, including SVM, KNN, decision trees, ensemble methods, and neural networks, using K-Fold Cross-Validation (K=4). Among all the classifiers, the *Coarse Tree* and *Bagged Tree* achieved the best performance (82% accuracy on the validation set). However, this performance was not as high as expected. With 123 training instances and 69 features, the low performance could be due to the high number of features. Therefore, we decided to reduce the model's complexity by selecting the 10 most significant features using the Feature Importance method. Most of these features represented the means of different channels and some bins of the intensity histograms.

We then retrained several classifiers using only these 10 features. As expected, we achieved higher accuracy (**90%**) but with a different classifier, the *Efficient Linear SVM*, which emerged as the best model.

Finally, we evaluated the best classifier on the test set, obtaining an accuracy of **86%**, which is very close to the validation accuracy.

4 Image Correction

The image correction process begins with determining the mean and variance of the luminance (Y channel) from correctly exposed images. These statistics serve as target values for correcting underexposed and overexposed images. The mean represents the average brightness level, while the variance measures the spread of brightness values. Accurate calculation of these values ensures that the gamma correction aligns the luminance of corrected images with that of correctly exposed images, achieving consistent and visually pleasing results.

4.1 Method 1: Preset Gamma Values

The first method, inspired by lecture material, involves applying gamma correction using *preset values*. Specifically, gamma values of 0.7 for underexposed images and 1.3 for overexposed images were chosen. The value 0.7 is used to brighten underexposed images, while 1.3 is used to darken overexposed images. These values were selected based on empirical observations and standard practices in image processing, where gamma values less than 1 brighten an image and values greater than 1 darken it. These corrections were applied to a small, random selection of images from each class to avoid computational overload and to ensure a diverse set of results. Random selection provides a comprehensive assessment of the correction process. This method is straightforward and provides a quick way to adjust image exposure, allowing for initial evaluation of the gamma correction's effectiveness.

4.2 Method 2: Brute-Force Approach

The second method employs a brute-force approach, systematically testing a range of gamma values to find the optimal one that minimizes the **Root Mean Square Error** (RMSE). RMSE is a metric that quantifies the difference between predicted and observed values, in this case, the luminance statistics of the corrected images compared to the target mean and variance. RMSE is chosen because it penalizes large errors more than smaller ones, providing a more sensitive measure of the correction quality. The brute-force method involves applying each gamma value to an image, calculating the RMSE for the corrected image's luminance, and selecting the gamma value with the lowest RMSE as the best correction. This exhaustive search ensures that the optimal gamma value is found, although it is computationally intensive, it guarantees the best possible adjustment for each image based on the defined range of gamma values.

4.3 Method 3: Optimization with fmincon

The third method uses the `fmincon` function for **optimization**, minimizing the RMSE to find the best gamma value. `fmincon` is chosen for its ability to handle constraints and provide robust solutions for nonlinear optimization problems. The process involves defining an objective function to calculate RMSE for a given gamma value, setting initial conditions and bounds for the gamma value, and using `fmincon` to iteratively adjust the gamma value, minimizing RMSE until convergence. This method is more efficient than brute-force, leveraging `fmincon`'s optimization capabilities to quickly find the best gamma value while ensuring the solution remains within a realistic and effective range. The constraints and bounds ensure that the gamma value remains practical and avoids extreme adjustments that could degrade image quality.

All images from the three classes were processed using the optimization method with `fmincon`. The corrected images were saved in a local directory for subsequent evaluation. This comprehensive processing ensures that every image benefits from the most precise gamma correction, aligning their luminance with the target values derived from correctly exposed images, thus preparing the dataset for further analysis and classification. By saving the processed images, we ensure that the corrections can be reviewed and used for additional testing and validation, solidifying the robustness of the gamma correction methods employed.

5 Discussion of the results

For the classification part, as mentioned in Section 3, our initial set of features did not yield high validation accuracy. This could be due to factors such as a small dataset or an overly complex model. Since increasing the dataset size was impractical, we decided to **simplify the model**.

In machine learning, it is generally recommended to have at least ten times as many training instances as features (Hastie, T., 2009). Using the Feature Importance method, applicable to tree-based models, we identified the *10 most significant features*, which included 5 means of different channels, 1 variance, and 4 bins of intensity histograms.

With this reduced set of features, the performance of all classifiers improved. The best performer, the *Efficient Linear SVM*, achieved a 90% accuracy on the validation set, which was an outstanding result. This good performance was confirmed on the test set, where the classifier achieved an accuracy of 86%, making only *2 incorrect predictions* out of 15. Thus, the final model for classifying our images proved to be highly robust.

Using the trained Efficient Linear SVM classifier and our developed optimization method, we achieved excellent results in classifying images corrected by the appropriate gamma correction. Out of 138 images, including the original correctly exposed images (used as a sanity check with applied gamma values), the classifier reached an accuracy of 83%. This demonstrates that our approach effectively handles new images corrected by our optimization algorithm, successfully completing both the classification and gamma optimization tasks.

Results confirmed by the additional metric that we have created for this: the entropy. Entropy measures the amount of information or randomness in an image, with higher entropy indicating more detail and texture. We selected entropy because gamma correction should reveal more details by reducing overly bright or dark areas. By comparing the entropy for the luminance channel (Y) of gamma-corrected images to the one of correctly exposed images, we assess whether the correction has enhanced image details appropriately. An acceptable range for entropy was defined as within two standard deviations of the mean, ensuring that the gamma correction maintained or improved image quality.

6 Conclusions

As explained in Sections 5 and 3, the performance achieved in the correction and classification of images indicates a robust and well-executed project. Furthermore, the new metric based on entropy, which we used to evaluate the gamma correction, offers an alternative perspective for assessing this task without relying on machine learning, even though it did not perform as well as the classifier.

Since we collected the images ourselves, there could be some mistakes or human biases that, if avoided, might have improved the performance. For instance, some images exhibit noise due to poor picture quality. To mitigate this issue, a larger dataset could have been collected, but this would require better hardware (e.g., GPUs), which was not available to us.

Future improvements could include collecting more data and using better hardware components. Additionally, exploring different, and potentially better, metrics for evaluating gamma correction could enhance the performance. While the entropy-based metric we used performed quite well, there may be more effective alternatives.

Overall, we are extremely satisfied with our work and the results achieved.

7 References

- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer.

- Gonzalez, R., & Woods, R. (2017). *Digital Image Processing* (4th ed., Global Edition). Pearson.
- Gonzalez, R. C., Woods, R. E., & Eddins, S. L. (2010). *Digital Image Processing Using MATLAB* (2nd ed.). Tata McGraw-Hill.