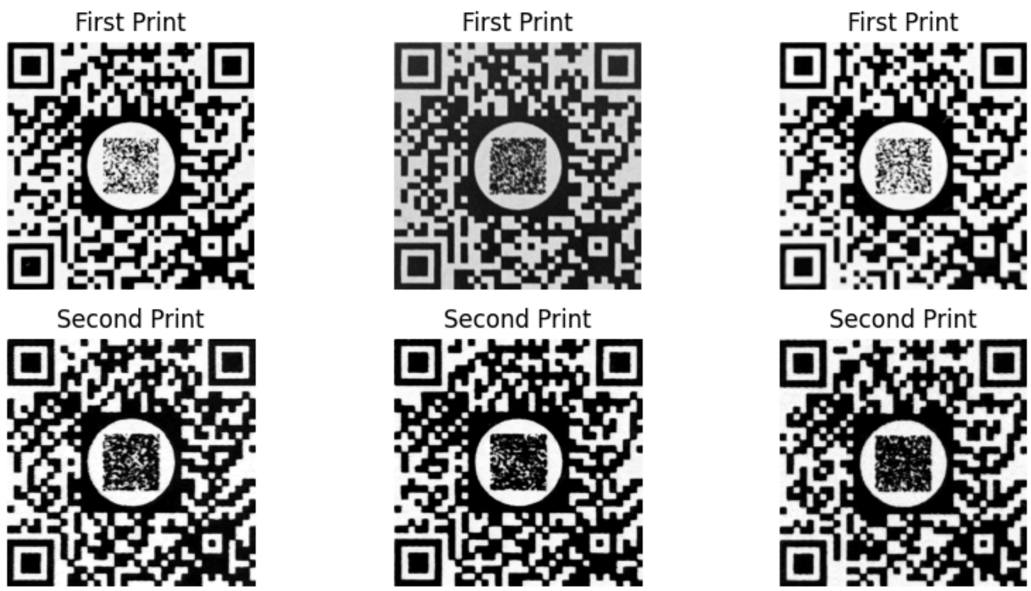
Detecting Original vs. Counterfeit Prints

1. Data Exploration and Analysis:

- 

The second print images are generally more smudged and have some loss in detail. For further analysis we can find out the statistics of the data.

-

```
Summary statistics by class:
      mean-intensity                                                  \
               count        mean        std        min         25%
label
first          100.0  121.614789  13.444310  97.930936  108.200246
second         100.0  105.162473   7.097267  87.411893  101.137706

                                           std-intensity             ...  \
              50%         75%         max          count        mean  ...
label                                                                 ...
first  128.782967  132.711928  137.937328          100.0  101.092796  ...
second 106.068688  110.039164  129.985738          100.0   97.242957  ...

                               sharpness                                     \
              75%         max     count        mean         std        min
label
first  113.617990  120.262216     100.0  326.806897  163.221078  38.748465
second 111.669844  116.939061     100.0  265.816423  104.715661  31.915233


              25%         50%         75%         max
label
first  256.300746  340.259857  437.394637  666.646374
second 230.207134  280.500048  334.816210  489.782457

[2 rows x 24 columns]
```

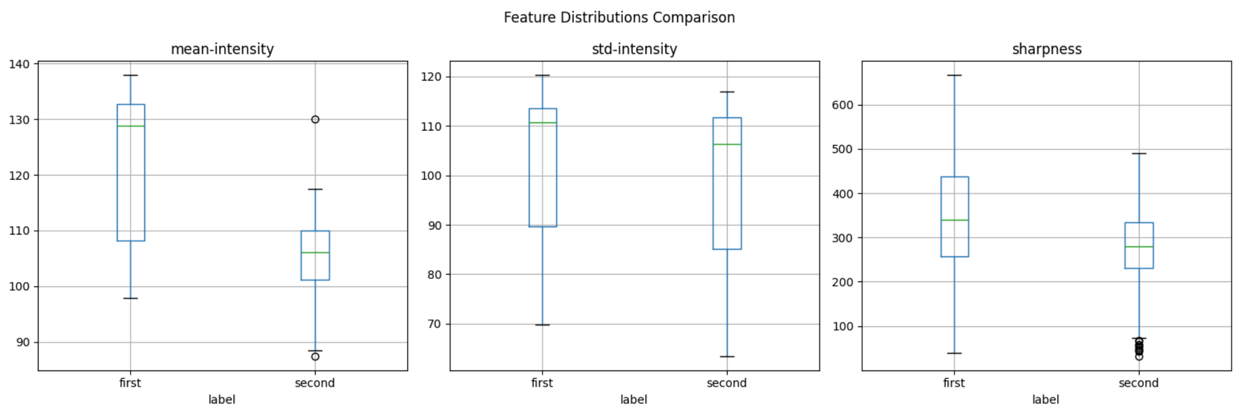
- We see that the first row has more mean-intensity and even more standard deviation.

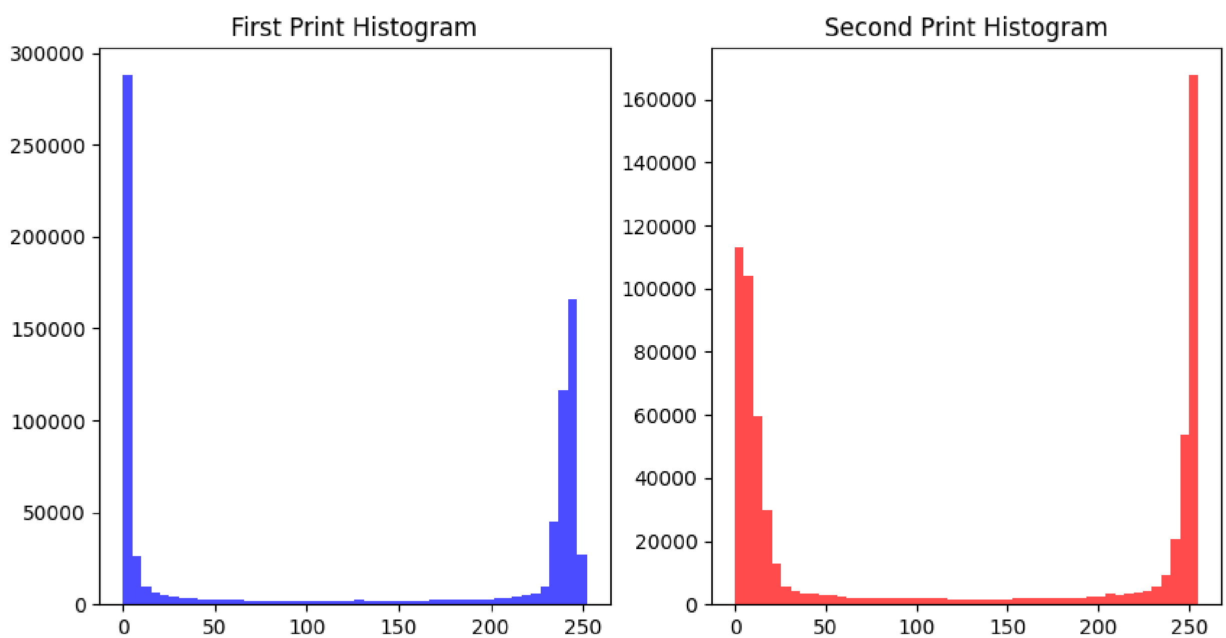
Explanation of Features:

1. **Mean (`mean`)**
    - The mean pixel intensity value of the image.
    - Represents the average brightness of the QR print.
    - **Higher mean** suggests a brighter image, while **lower mean** indicates a darker image.
2. **Standard Deviation (`std`)**
    - Measures the spread of pixel intensities.
    - **High standard deviation** means the image has more contrast (sharp variations in pixel values).

- **Low standard deviation** suggests a more uniform image.

3. **Sharpness (`Laplace Variance`)**
   - Calculated using the variance of the Laplacian filter.
   - **Higher values** mean the image is sharp, while **lower values** indicate blur.





```
Statistical significance (t-tests):
mean: t-stat = 10.82, p-value = 0.0000
std: t-stat = 1.66, p-value = 0.0977
sharpness: t-stat = 3.15, p-value = 0.0019

Logistic Regression Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.75      0.79        20
           1       0.77      0.85      0.81        20

    accuracy                           0.80        40
   macro avg       0.80      0.80      0.80        40
weighted avg       0.80      0.80      0.80        40
```

This indicates that the features (mean, std, sharpness) are somewhat effective at distinguishing between original and counterfeit prints. So for better results we need somewhat more complex features that can capture the relevant information of the images to make classification.

2. Feature Engineering :

1. **Global Image Properties**:
    - **Mean & Standard Deviation**: Captures overall brightness and contrast variations between genuine and counterfeit prints. Second prints tend to have slight intensity variations due to scanning noise and printer inconsistencies.
    - **Sharpness (Laplacian Variance)**: Helps detect blurriness, which might indicate printing artefacts or lower-quality reproduction. Scanning and reprinting introduce slight blurring and loss of fine details.
2. **Texture Analysis (GLCM Features)**:
    - **Contrast**: Measures intensity variation, distinguishing smooth vs. rough textures in different prints. Counterfeit prints often lose high-frequency texture details due to resampling.
    - **Homogeneity**: Quantifies how similar pixel intensities are; genuine prints may have more uniform patterns. Reprinting processes often smooth out fine texture

details, reducing textural diversity.

- **Energy**: Represents textural uniformity, which can differ based on print artefacts. A measure of textural uniformity; scanned copies may introduce random noise affecting uniformity.
- **Correlation**: Evaluates pixel relationships, useful for detecting inconsistencies in counterfeit prints. First prints maintain stronger correlations, while second prints exhibit weakened correlation patterns.

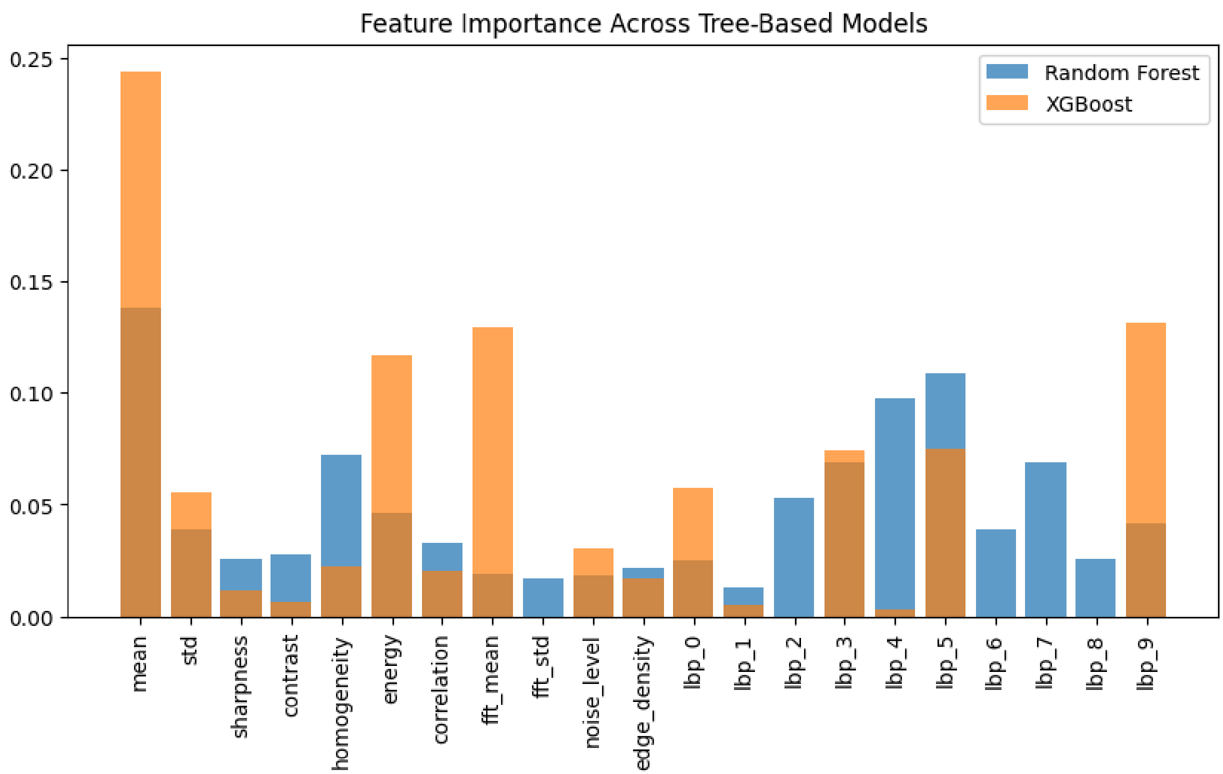
3. **Frequency Domain Features (Fourier Analysis)**:
   - **FFT Mean & FFT Standard Deviation**: Analyzes spatial frequency distribution, highlighting differences in print resolution and structure.
4. **Print Artifacts & Noise Features**:
   - **Noise Level**: Measures standard deviation of pixel differences after Gaussian blurring; helps detect inconsistencies in printing techniques. Second prints tend to have higher noise levels compared to original first prints.
   - **Edge Density**: Uses Canny edge detection to capture the presence of fine details, which may be lost in counterfeit prints. First prints will have a more defined frequency structure, whereas second prints lose high-frequency details due to resolution limitations and resampling.
5. **Local Pattern Features (LBP - Local Binary Patterns)**:
   - **LBP Histogram**: Encodes micro-texture patterns, useful for identifying fine print differences and degradation in counterfeit copies. Captures micro-texture changes that occur due to loss of fine details during scanning and reprinting.

These features collectively enhance the ability to differentiate between original and counterfeit prints by considering **global, texture, frequency, and local-level artefacts**.

These results with these features are better, but still, since the data was of only 200 images, it was **overfitting**. So, I had to reduce the lbp bins to only 10.

```
Model Performance Comparison:
                    Model  Accuracy  F1-Score  Mean CV Accuracy
0     Logistic Regression  0.983333  0.983607             0.885
1  Support Vector Machine  0.983333  0.983607             0.870
2           Random Forest  0.966667  0.966667             0.990
3     K-Nearest Neighbors  0.983333  0.983051             0.655
4                 XGBoost  0.916667  0.912281             0.985
```

Feature importance of these features:



Model Development

- **Traditional Computer Vision + Machine Learning Approach**
    - Utilized handcrafted features such as GLCM, LBP, FFT, and Edge Density.
    - Trained a Random Forest Classifier to distinguish between original and counterfeit QR codes.
    - Applied cross-validation to ensure model generalization and robustness.
    - Effective for capturing structured image properties with lower data requirements.

- **Deep Learning-Based Approach (CNN)**
  - Designed a CNN with convolutional layers, adaptive average pooling, and fully connected layers.
  - Applied data augmentation (rotation, affine transform, sharpness adjustment) for better generalization.
  - Used Binary Cross-Entropy Loss and Adam optimizer for training.
  - Automatically learns patterns and distortions in QR codes, detecting subtle print artifacts.
- **Validation Strategies**
  - Used 5-fold cross-validation for the Random Forest model to assess reliability.
  - Split dataset into 80% training and 20% testing for CNN to evaluate model performance.
  - Analyzed misclassified samples to understand model limitations and improvement areas.
- **Implementation Choices & Reasoning**
  - Random Forest leverages structured features for interpretability and efficiency with small datasets.
  - CNN enables feature learning directly from raw images, handling complex distortions better.
  - Combining both approaches can improve overall classification accuracy.

Evaluation and Results

- **Model Performance Metrics**
  - The CNN achieved a **final accuracy of 97.5%**.
  - Evaluation included **accuracy, precision, recall, and F1-score** to assess performance comprehensively.
- **Misclassification Analysis**
  - The CNN misclassified **only one sample** from the test set.
  - Possible reasons include **image noise, print artifacts, or borderline feature values**.
- **Comparison of Approaches**
  - The CNN **outperformed** the traditional machine learning pipeline by leveraging deeper feature representations.

    - Machine learning models like **Random Forest** were still useful for feature interpretability.

- **Handling Misclassified Samples**

    - **Ensemble Learning:** Combine predictions from multiple models to reduce errors.
    - **Feature Refinement:** Improve feature extraction techniques to capture more distinguishing details.
    - **Threshold Tuning:** Adjust decision boundaries in the CNN to minimize borderline misclassifications.