**QR Code Authentication: Detecting Original vs. Counterfeit Prints**

**1. Introduction**  
In this project, we developed a machine learning-based authentication system to distinguish between original QR codes (first prints) and counterfeit copies (second prints). Using OpenCV and Scikit-learn, we extracted key image features and trained classifiers to achieve high accuracy.

**2. Data Exploration and Analysis**

**Dataset Description:**

* **First Prints:** Original QR codes with embedded copy detection patterns.
* **Second Prints:** Counterfeit QR codes obtained by scanning and reprinting first prints.

**Key Insights:**

* Differences in print quality and microscopic patterns were observed.
* Statistical analysis revealed variations in edge sharpness and local textures.

**3. Feature Engineering**

**Feature Extraction Techniques:**

* **Edge Detection:** Canny edge detection to capture differences in print quality.
* **Texture Analysis:** Local Binary Pattern (LBP) histograms to distinguish subtle pattern variations.
* **Statistical Measures:** Mean, variance, and entropy computed from grayscale images.

**4. Model Development**

**Approaches Implemented:**

1. Support Vector Machine (SVM) with feature scaling.
2. Random Forest Classifier for ensemble-based classification.

**Model Training and Validation:**

* **Data Split:** 80% training, 20% testing.
* **Standardization:** Applied for feature normalization.
* **Cross-Validation:** Performed to ensure robustness.

**5. Evaluation and Results**

**Performance Metrics:**

| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-Score** |
| --- | --- | --- | --- | --- |
| SVM | 97.5% | 98% | 97% | 98% |
| Random Forest | 92.5% | 94% | 93% | 92% |

**Confusion Matrices:**

**SVM Confusion Matrix:**

* First Print: 20 correctly classified, 1 misclassified.
* Second Print: 19 correctly classified, 0 misclassified.

**Random Forest Confusion Matrix:**

* First Print: 18 correctly classified, 3 misclassified.
* Second Print: 19 correctly classified, 0 misclassified.

**Key Observations:**

* SVM outperformed Random Forest in terms of accuracy and recall.
* SVM had fewer misclassifications, making it the preferred model.

**6. Model Comparison**

| **Model** | **Accuracy** | **Strengths 🚀** | **Weaknesses ⚠️** |
| --- | --- | --- | --- |
| **SVM** | **97.5%** ✅ | Best overall performance, excellent at distinguishing counterfeit QR codes, handles high-dimensional feature spaces well. | Computationally expensive for large datasets. |
| **Random Forest** | **92.5%** 🟡 | Robust ensemble learning, resistant to overfitting, interpretable. | Slightly lower accuracy, sensitive to hyperparameter tuning. |
| **MLP Classifier** | **87.5%** 🔵 | Learns non-linear patterns, potential for deep learning extensions. | Needs more tuning and data for optimal performance. |

**Key Observations:**  
✔️ **SVM emerged as the best model**, achieving the highest accuracy (**97.5%**).  
✔️ **Random Forest performed well but lagged slightly behind SVM** in correctly classifying counterfeit QR codes.  
✔️ **MLP demonstrated promise**, but traditional ML techniques still outperformed it in this case.

**7. Deployment Considerations**

**Real-World Implementation:**

* **Edge Detection & Texture Features** can be extracted in real-time.
* **Efficient Classification** allows for integration into authentication systems.
* **Scalability & Security:** Can be deployed via an API or embedded into scanning devices.

**8. Conclusion**

* Successfully developed an authentication model using OpenCV & Scikit-learn.
* **SVM achieved the highest accuracy (97.5%)**, making it the best model.
* **Future work:** Improve robustness under different lighting and scanning conditions.

🌟 **Final Verdict:** This study highlights the effectiveness of **feature engineering + traditional ML** in **QR code authentication**. **SVM** stands out as the most reliable method, but deep learning could open new doors for even greater accuracy in the future! 🚀