🔍 Tamper Detection in Academic Credentials

Prepared by: B Sai Sri Harshitha

Tools Used: Python, PyPDF2, pdf2image, OpenCV, Tesseract OCR, scikit-learn

1. Objective

The objective of this project is to develop a prototype system that can **automatically detect tampering** in academic documents such as:

- Degree Certificates
- Academic Transcripts
- Professional Certifications

The system detects anomalies based on:

- PDF metadata inconsistencies
- Visual layout differences
- OCR text deviations
- Statistical outliers in document behavior

2. Methodology

To ensure comprehensive tamper detection, we applied a **multi-layered approach** combining metadata analysis, visual comparison, OCR-based text matching, and machine learning-based anomaly detection.

A. PDF Metadata Analysis

- Tool: PyPDF2
- Process:
  - Extracted metadata fields such as /CreationDate, /ModDate, /Author, /Producer.
  - Flagged documents where the modification date was later than the creation date or significantly deviated.
- Sample Code:

python

```
from PyPDF2 import PdfReader
reader = PdfReader("degree_tampered.pdf")
metadata = reader.metadata
if metadata.get('/ModDate') != metadata.get('/CreationDate'):
    print("⚠️ Metadata tampering suspected.")
```

B. Layout Analysis using OpenCV

- Tools: pdf2image, OpenCV, skimage.metrics.structural_similarity
- Process:
  - Converted the first page of PDFs to images using pdf2image.
  - Compared images (original vs tampered) using **Structural Similarity Index (SSIM)**.
  - Documents with SSIM score below **0.95** were flagged as layout-tampered.
- Visualization: Highlighted mismatched areas for visual confirmation.

C. OCR-Based Text Comparison

- Tools: pytesseract, Pillow
- Process:
  - Used **Tesseract OCR** to extract text from document images.

- ○ Comparing full text content between original and tampered versions.
- ○ Flagged discrepancies such as changed names, dates, grades, etc.

- Challenges: Minor font or format differences can introduce noise, handled with preprocessing.

D. Anomaly Detection (Bonus Task)

- Tool: scikit-learn (IsolationForest)
- Features extracted:
  - mod_gap_days: Days between creation and modification date
  - layout_score: SSIM comparison score
  - ocr_text_length: Number of characters in OCR output
  - metadata_flag: Binary flag if metadata was suspicious
- Process:
  - Trained an Isolation Forest on “normal” documents
  - Flagged statistically deviant documents as anomalies

3. Results

Document	**Metadata Tamperec**	**Layout Anom**	**OCR Differen**	**Anomaly Flag**
degree_tampered.pdf	✅ Yes	✅ Yes	❌ No	✅ Yes
transcript_fake.pdf	❌ No	❌ No	✅ Yes	✅ Yes
cert_fake.pdf	❌ No	✅ Yes	✅ Yes	✅ Yes
degree_original.pdf	❌ No	❌ No	❌ No	❌ No

The system successfully identified multiple types of tampering across different document types, confirming the reliability of a multi-pronged approach.

4. Challenges

- **Metadata Issues**: Not all PDFs have editable metadata; some are encrypted or flattened scans.
- **Layout Sensitivity**: Layout comparison accuracy drops with varying scan resolution of image noise.
- **OCR Noise**: Fonts, misalignment, and compression artifacts introduce OCR inaccuracies.
- **Baseline Dependence**: Anomaly detection requires a clean baseline of untampered documents to be effective.

5. Suggestions & Future Work

- 🧾 **Blockchain Signatures**: Embed certificates with hash or blockchain-backed authenticity markers.
- 🔍 **Labeled Dataset Expansion**: Collect real-world tampered and authentic samples for training.
- 🤖 **Supervised ML Models**: Use classifiers (e.g., RandomForest, XGBoost) with labeled data for higher precision.
- 🔐 **QR / Watermark Verification**: Add and validate secure QR codes or invisible watermarks.

6. Conclusion

This prototype shows that **automated tamper detection** in academic credentials is **feasible** using:

- PDF metadata inspection
- Visual layout comparison
- OCR text analysis
- Machine learning-based anomaly detection

The integration of these modules provides a **robust framework** for verifying academic documents in real-world applications such as recruitment, admissions, and background verification. The system can be scaled and enhanced with better datasets and integrations.