Face - Counter Spoofing based on BENFORD'S Statistical Compliance

A Thesis submitted in fulfillment of the requirements for the Degree

of

Master of Technology

in

Signal Processing and Machine Learning

by

CHERUPELLI ANVESH

Roll No: 234102312

Under the supervision

of

Dr. Kannan Karthik

Department of Electronics and Electrical Engineering

INDIAN INSTITUTE OF TECHNOLOGY

GUWAHATI , ASSAM - 781039

JUNE 2025

DECLARATION

This is to declare that the work which is being presented in the thesis entitled named **Face - Counter spoofing based on Benford's Statistical Compliance** submitted to the **Indian Institute of Technology Guwahati** for the granting of the **Master of Technology** degree is a genuine work completed under **Dr. Kannan Karthik**

I here by swear that this thesis is entirely original and that no submissions are made to obtain credit toward another degree or professional certification. I also want to clarify that, to the best of my understanding, there is no plagiarism in this report.

Cherupelli Anvesh
Roll.No. 234102312
Dept. of Electronics and Electrical Engineering
IIT Guwahati, Assam, India -781039
June 2025

CERTIFICATE

This is to certify that the work contained in the thesis entitled **Face - Counter spoofing based on Benford's Statistical Compliance**, *by* **Cherupelli Anvesh (Roll No. 234102312)**, which has been carried out in the Department of Electronics and Electrical Engineering, Indian Institute of Technology Guwahati under my supervision during the **academic year 2024–2025**, and has not been submitted elsewhere for the award of any degree.

Dr. Kannan Karthik
Associate Professor
Dept. of Electronics and Electrical Engineering
Indian Institute of Technology Guwahati
Guwahati, Assam, India- 781039
June 2025

Acknowledgement

I am immensely thankful to **Dr.Kannan Karthik**, my supervisor, for his unwavering guidance, support, and motivation throughout the process of composing this report. His consistent supervision and valuable insights steered me in the right direction whenever I faced challenges. I am sincerely grateful for his dedicated efforts in proofreading and correcting my work.

I extend my heartfelt appreciation to my parents, as well as my friends, for their unparalleled support and continuous encouragement. Their unwavering backing has been instrumental in making this achievement possible. I am deeply thankful to them as this accomplishment would not have been attainable without their presence and support.

Abstract

This work addresses the critical challenge of face counter spoofing, a vital component for securing modern face recognition systems against sophisticated presentation attacks. Although face recognition offers unparalleled convenience and efficiency in biometric authentication, its vulnerability to various spoofing methods, including print, replay, and mask attacks, poses significant security and privacy risks.

In applications involving data coming from a mixture of diverse sources, it was observed that the first digit distributions (FDD) of this measurement data closely follow Benford's law.This holds for most natural mixtures.In the context of face spoofing application,it was therefore anticipated that natural face presentations and subsequent localized measurements are more likely to follow Benford's law as compared to images obtained via planar print spoofings.

The proposed system first extracts DCT coefficients from genuine and spoofed face images. Subsequently, the first significant digits of these coefficients were analyzed to compute their respective Benford distributions. By quantifying the divergence from the natural Benford distribution, a clear statistical distinction between real and fake images is established. Our experiments, conducted on diverse datasets encompassing various spoofing scenarios, demonstrate the efficacy of this Benford's law-based analysis in accurately classifying presentation attacks. This approach offers a promising direction for developing highly generalizable face anti-spoofing solutions, relying on fundamental statistical irregularities rather than specific visual artifacts.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Imagine how often we now use our faces to unlock phones, get into buildings, or even pay for things online. It's super handy because you don't have to touch anything, it's quick, and pretty much everyone has a face!

But here's the catch: because it's so common, sneaky people are always trying to trick these face systems. They use "spoofing attacks" – basically, presenting something that looks like your face, but isn't actually you. This could be anything from a printed photo, to a video played on a screen, or even a fancy 3D mask that's made to look like a real person. These tricks can fool even the most advanced face recognition systems, which is a big problem for our privacy and security.

In this project, we are digging deep into this problem. We are looking at different ways faces are spoofed – like using photos, videos, or masks – and trying to combine what we see with clever ways to categorize genuine and fake attempts. Our aim is to detect whether an exposed image of a person is real or spoofed?

Figure 1.1: Types of different attacks
a) Real Image, b) Printed flat photo, c) Eye cut photo, d) Warped photo, e) Video playback, f) Synthetic mask, and g) Paper cut mask.

Types of Spoofing attacks

- **Planar spoofing** Fig 1.1 & Fig 1.2 shows [1][2] that this involves presenting 2D models such as printed photocopy or showing on a tablet piece with printed photographs to a face recognition system.
- **Prosthetic spoofing** involving 3D models such as 3D face mask made up of silicone, latex, or fabric to match with real face textures.

Difference between Natural image & Spoof image

Feature Type	**Natural Image**	**Spoofed Image**
Origin	Captured directly from the real world by a camera.	Created or altered using software (e.g., Photoshop, AI generative models, 3D rendering).
Physical Properties	Exhibits consistent and realistic physical properties (e.g., light, shadow, reflections, depth, texture)	Often shows inconsistencies or anomalies in physical properties.
Lighting	Consistent light sources, natural reflections, and shadows that align with the environment	Abnormal lighting, mismatched reflections, inconsistent shadow directions, or missing shadows.
Texture & Detail	Rich, varied, and natural textures; fine details are generally consistent	Repetitive patterns, overly smooth or unnatural textures (e.g., in skin, hair).
Color Fidelity	True-to-life colors, Natural Color diversity	Color reproduction loss, or unnatural color shifts.

Table 1.1: Comparison of Natural and Spoofed Images

Figure 1.2: Natural real faces(top row), Spoof variations(bottom row)

Figure 1.3: Natural real faces(top row), Spoof variations(bottom row)

Chapter 2

Literature Review

Face counter-spoofing methods can broadly be categorized into **hardware-based** and **software-based** approaches.

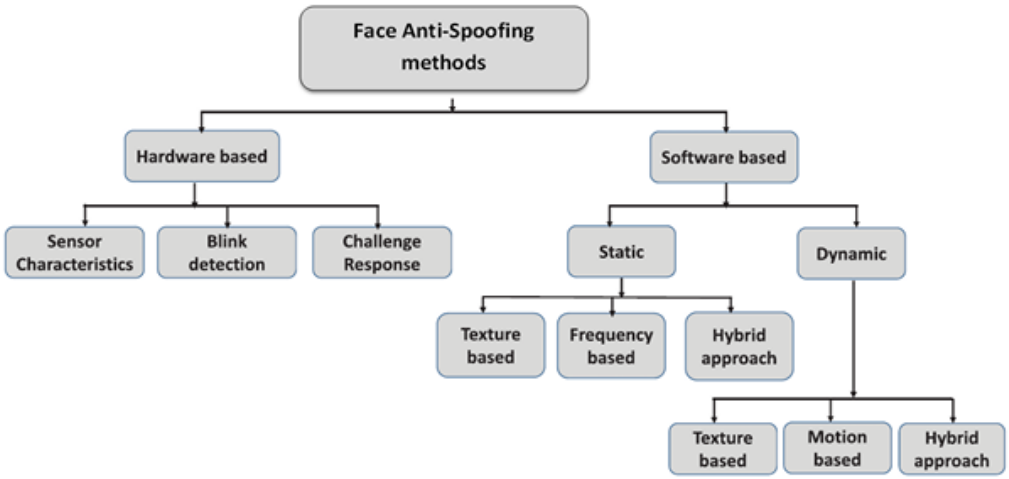

Figure 2.1: Classification of Anti-Spoofing Methods [1]

2.1 Hardware-Based Methods

These methods require additional sensors to detect physical or biometric cues beyond RGB images:

- **Depth sensors:** Use infrared or structured light to measure 3D facial geometry[3].
- **Thermal imaging:** Differentiates real skin temperature from spoofed materials[4].
- **Multispectral imaging:** Analyzes reflectance across different wavelengths.

2.2 Software-Based Methods

These techniques analyze RGB images or videos using machine learning or deep learning:

2.2.1 Texture-Based Approaches

- **Protection of 2D face identification systems against spoofing attacks** (Edmunds, Taiamiti), HSI-LBP color texture descriptor(method). Print, mobile and ipad (attack type) case.[5]

- **Entropy-Based Face Recognition and Spoof Detection for Security Applications** (Pujol, Francisco A.) CASIA FASD [6]

- **Face Anti-Spoofing using Texture-Based Techniques and Filtering Methods**
  (Md. Rezwan Hasan and S. M. Hasan Mahmud)
  Photo attack NUAA Photograph Imposter database [7].

2.2.2 Deep Learning-Based Approaches

- **Integration of image quality and motion cues for face anti-spoofing**:
  Multi-task CNN that jointly estimates spoofing, depth, and rPPG (remote photoplethysmography) signals[8].

2.2.3 Observations & Gaps

- Many models overfit to specific attack types or datasets and fail under cross-dataset evaluation.

- Deep models require large and diverse training data; limited availability of mask or high-quality replay data is a challenge.

- Depth and motion cues are robust but require either complex models or special hardware.

Chapter 3

Motivation & Importance

With the widespread deployment of facial recognition in smartphones, banking apps, surveillance systems, and border control, face spoofing has emerged as a significant security threat. Presentation attacks using printed photos, replayed videos, or even hyper-realistic 3D masks can deceive conventional face recognition systems. Therefore, face anti-spoofing also known as Presentation Attack Detection (PAD) is crucial for:

- **Enhancing biometric security:** Preventing identity theft and unauthorized access.
- **Preserving trust in facial recognition systems:** Especially in high-security environments.
- **Reducing financial fraud and impersonation risks:** In banking, e-commerce, and government services.
- **Compliance with biometric data regulations:** Where liveness detection is often a legal requirement.

Chapter 4

Problem Statement

An alternative and promising direction lies in exploiting statistical irregularities introduced during the creation or manipulation of spoofed media. One such approach uses Benford's law, a mathematical principle that describes the expected distribution of first digits in natural data sets. When images are captured or manipulated, especially through compression, printing, or digital replay, these processes may distort the natural statistical properties of pixel intensities or transform coefficients, leading to deviations from Benford's distribution.

- **CLAIM :** " This work introduces the premise that when images undergo processes such as capture, manipulation, compression, printing, or digital replay, these operations can distort the natural statistical properties of pixel intensities or transform coefficients. This distortion may lead to deviations from Benford's distribution, providing a statistical indicator that the media has been manipulated or is not authentic".

- In support of evaluating **CLAIM**, we proposed a methodology which is described in Chapter 5.

- So, we evaluated our claim to be true by showing that when an image undergoes processes such as digital capture attacks, manipulations can distort the natural statistical properties of pixel intensities or transform coefficients.

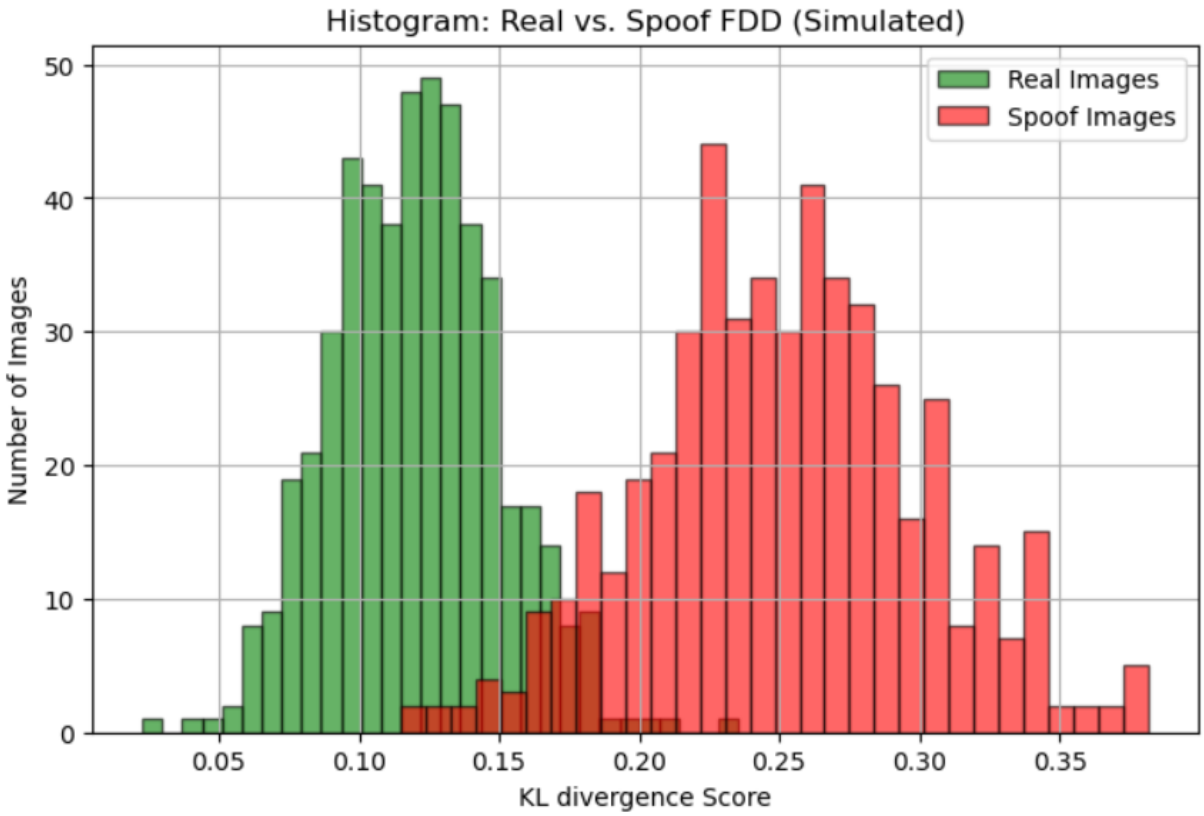

Figure 4.1: Histogram of FDD Score

$$D_{\mathrm{KL}}(P_{\mathrm{obs}} \| P_{\mathrm{ben}}) = \sum_{d=1}^{9} P_{\mathrm{obs}}(d) \cdot \log\left(\frac{P_{\mathrm{obs}}(d)}{P_{\mathrm{ben}}(d)}\right) \tag{4.1}$$

This score measures how much the observed distribution P deviates from Benford's Law Q. A higher FDD score indicates a greater deviation and potentially unnatural or manipulated content.

Summary of KL Divergence Score :

- A comparative histogram of KL divergence-based FDD scores between real and spoofed images shows clear separability. Real images tend to have lower FDD scores, indicating a closer alignment with Benford's law.
- In contrast, spoof images demonstrate significantly higher FDD scores, suggesting statistical irregularities. This contrast provides a reliable metric to distinguish authentic faces from false claims

Chapter 5

Methodology

Data Preprocessing

This work focuses on analyzing face images for spoof detection using statistical properties derived from local image regions. Two types of features are explored: Discrete Cosine Transform (DCT) coefficients and Histogram of Oriented Gradients (HOG) descriptors. The primary aim is to detect deviations from natural image statistics using Benford's Law, which helps differentiate between real and spoofed facial images.

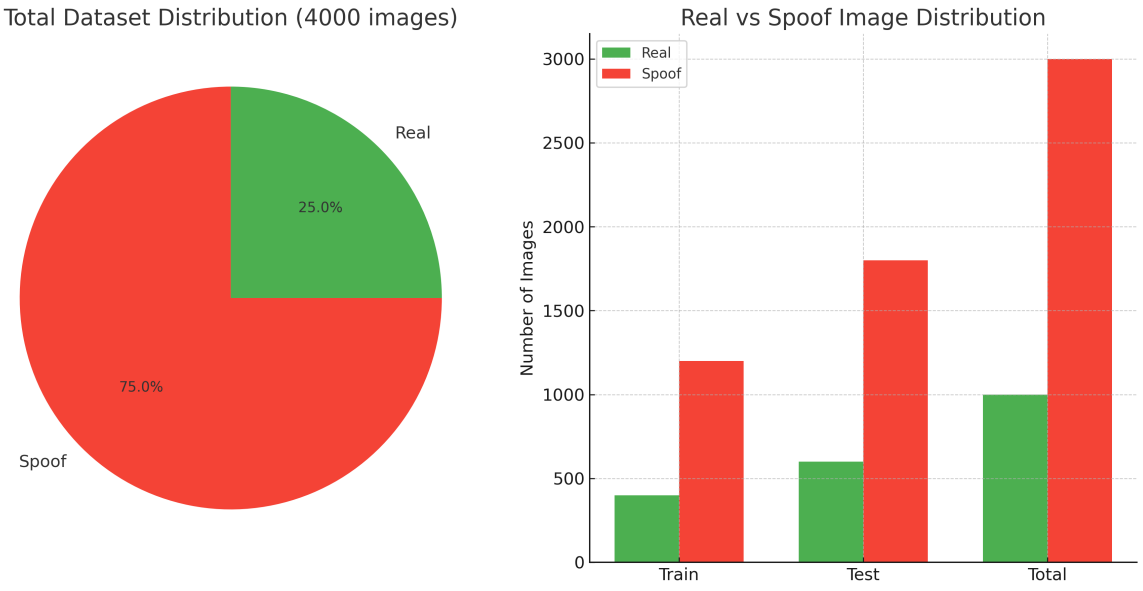

Figure 5.1: CASIA FASD Dataset

1. Feature Extraction Process

Image Segmentation(Patch division): For face images, focus the DCT analysis on specific regions of interest (e.g. the face region itself) rather than the entire background of the image.

2. DCT Feature Extraction

- The 2D DCT is applied to each image patch to convert spatial information into frequency components.From the DCT output, the AC coefficients (excluding the DC component) are selected.
- A Benford deviation score is computed using distance metrics such as Euclidean distance between the empirical and theoretical Benford distributions.

5.1 Algorithms

5.1.1 DCT Feature Extraction from 8×8 Blocks

Algorithm 1 DCT Block-Based Feature Extraction

Input: Image data img, block size b
Output: DCT feature vector

Convert img to grayscale
Get height h and width w of grayscale image
Initialize empty feature list `features`
for $i = 0$ to $h - b$ step b **do**
    for $j = 0$ to $w - b$ step b **do**
        Extract block $B = img[i : i + b,\ j : j + b]$
        Compute DCT along rows: $D1 = DCT(B^T)^T$
        Compute DCT along columns: $D2 = DCT(D1)$
        Take absolute and flatten: $F = |D2|.flatten()$
        Append F to `features`
    end for
end for
return `features`

Algorithm Procedure:

1. **Convert to Grayscale**

   - Color is not necessary for DCT since structural and textural patterns can be analyzed in grayscale.

2. **Extract Block**

   - B = img[i:i+b, j:j+b]: A square patch/block of size b×b.

3. **Apply 2D DCT**

   - B = img[i:i+b, j:j+b]: A square patch/block of size b×b.

$$F(u,v) = \frac{1}{4}\alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

$$\alpha(k) = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0 \\ 1, & \text{otherwise} \end{cases}$$

4. **Collect All AC Coefficients**

   - From each block, discard the DC coefficient C(0,0)C(0,0), From each block, discard the DC coefficient $C(0,0)$ and retain the AC coefficients:

$$A = \{a_1, a_2, ..., a_N\}$$

5. **First Digit Extraction(Take Absolute and Flatten):**

$$d_i = \text{first\_digit}(|a_i|), \quad \forall i = 1 \text{ to } N$$

   The resulting digit sequence is:

$$D = \{d_1, d_2, ..., d_N\}$$

   - |D2|: Take the absolute values of DCT coefficients (removing the sign, keeping the magnitude).

- flatten(): Convert a 2D array to a 1D vector.

6. **Compute Observed Frequency Distribution**

   We form a normalized histogram of the first digits from D to get the observed probability distribution:

$$P_{\text{obs}}(d) = \frac{\text{count of } d \text{ in } D}{N}, \quad \text{for } d \in \{1, 2, ..., 9\}$$

7. **Compute Benford Distribution**

   According to Benford's Law, the expected probability of the first digit d is:

$$P_{\text{ben}}(d) = \log_{10}\left(1 + \frac{1}{d}\right)$$

8. **Compute KL Divergence**

   The KL divergence between the observed and expected distributions is computed as:

$$D_{\text{KL}}(P_{\text{obs}} \| P_{\text{ben}}) = \sum_{d=1}^{9} P_{\text{obs}}(d) \cdot \log\left(\frac{P_{\text{obs}}(d)}{P_{\text{ben}}(d)}\right)$$

   This score quantifies the deviation of the real data from Benford's distribution.

Image Type	**Avg. KL Score**	**Compliance**
Real Image	0.012	High
Spoof Image	0.076	Low

5.1.2 HOG Feature Extraction

Algorithm 2 HOG Feature Extraction

Input: Image data *img*
Output: HOG feature vector

1: Convert *img* to grayscale
2: Set HOG parameters:
- orientations = 9
- pixels per cell = 8×8
- cells per block = 2×2

3: Compute HOG descriptor H using above parameters
4: Flatten H to obtain feature vector
5: **return** H

Algorithm Procedure:

1. **Convert to Grayscale**
   - Like with DCT, gradient and edge patterns are well captured in grayscale.
   - Reduces dimensionality and complexity.
2. **Set HOG Parameters**
   - orientations = 9: Number of bins for edge directions (0° to 180° split into 9 angles).
   - pixels per cell = 8×8: Each cell contains 8×8 pixels.
   - cells per block = 2×2: Each block spans 2×2 cells, with normalization over the blocks.
3. **Compute HOG Descriptor**
   - Compute gradients in the x and y directions using filters (Sobel Operator).

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

4. **Gradient Computation**

   Given a grayscale image $I(x, y)$, compute gradients:

$$G_x(x, y) = I(x + 1, y) - I(x - 1, y), \quad G_y(x, y) = I(x, y + 1) - I(x, y - 1)$$

5. **Magnitude and Orientation**

   Compute gradient magnitude and orientation angle:

$$M(x, y) = \sqrt{G_x^2(x, y) + G_y^2(x, y)}$$

$$\theta(x, y) = \arctan\left(\frac{G_y(x, y)}{G_x(x, y)}\right)$$

$$\theta(x, y) = \tan^{-1}\left(\frac{G_y(x, y)}{G_x(x, y)}\right)$$

6. **Cell Histograms**

   Divide the image into 8×8 pixel cells. For each cell, compute histogram of orientations:

$$H_c = \{h_1, h_2, ..., h_n\}$$

   where h_i is the magnitude sum for orientation bin i in cell c.

7. **Block Normalization**

   Group 2×2 cells into blocks and concatenate their histograms into vector v. Normalize:

$$v' = \frac{v}{\sqrt{\|v\|_2^2 + \epsilon^2}}$$

8. **Final Feature Vector**

   Concatenate all normalized block vectors to form:

$$F = \text{concat}(v_1', v_2', ..., v_k')$$

9. **Benford Analysis on Magnitudes**

   Extract first digits from all gradient magnitudes:

$$D = \{d_1, d_2, ..., d_N\}, \quad d_i = \text{first\_digit}(M_i)$$

   Compute histogram:

$$P_{\text{obs}}(d) = \frac{\text{count of } d \text{ in } D}{N}$$

   Benford distribution:

$$P_{\text{ben}}(d) = \log_{10}\left(1 + \frac{1}{d}\right)$$

10. **KL Divergence**

$$D_{\text{KL}}(P_{\text{obs}} \| P_{\text{ben}}) = \sum_{d=1}^{9} P_{\text{obs}}(d) \cdot \log\left(\frac{P_{\text{obs}}(d)}{P_{\text{ben}}(d)}\right)$$

   This score quantifies the deviation of the real data from Benford's distribution.

Image Type	**Avg. KL Score (HOG)**	**Compliance**
Real Image	0.018	High
Spoof Image	0.092	Low

Chapter 6

Results

To assess the naturality of facial images, we calculated the deviation of their distributions of DCT coefficients from Benford's law. Real (original) images are expected to follow this law due to the inherent statistical regularities in natural scenes.

In contrast, spoof images, typically created by printing and recapturing photographs, introduce unnatural artifacts that distort frequency patterns. In our experiment, the original image exhibited a low Benford deviation (e.g., 0.0342), indicating high naturality. The spoof image showed a significantly higher deviation (e.g., 0.1259), suggesting a loss of natural structure.

This validates the use of Benford-based deviation as a reliable indicator to distinguish between real and spoof facial images.

6.1 Naturality Check on CASIA CALIBERATION

Mean Spatial distance between Sub-Patches

- Given an image patch of size 32×32
- we divide it into non-overlapping sub-patches of size 8×8.
- Each sub-patch is denoted as $\mathbf{B}_k \in \mathbb{R}^{8\times8\times3}$, where k indicates the block index, and the third dimension corresponds to the RGB color channels.
- We define the Euclidean distance between two neighboring blocks $\mathbf{B}_k$ and $\mathbf{B}_l$ (either to the right or bottom of $\mathbf{B}_k$) as:

$$d(\mathbf{B}_k, \mathbf{B}_l) = \sqrt{\sum_{i=1}^{8}\sum_{j=1}^{8}\sum_{c=1}^{3}\left(\mathbf{B}_k(i,j,c) - \mathbf{B}_l(i,j,c)\right)^2} \tag{6.1}$$

Let N be the total number of valid neighboring block pairs (right and bottom neighbors). Then, the mean spatial distance across all such pairs is computed as:

$$D_{\text{mean}} = \frac{1}{N} \sum_{(k,l)\in\text{Neighbors}} d(\mathbf{B}_k, \mathbf{B}_l) \tag{6.2}$$

6.2 Check with DCT Feature Vectors + Benford's law

```
Observed FDD: [0.44678991 0.18923308 0.11503872 0.07544342 0.05108668 0.03934549
 0.03322508 0.02860355 0.02123407]
KL Divergence from Benford: 0.0669249512214237
```

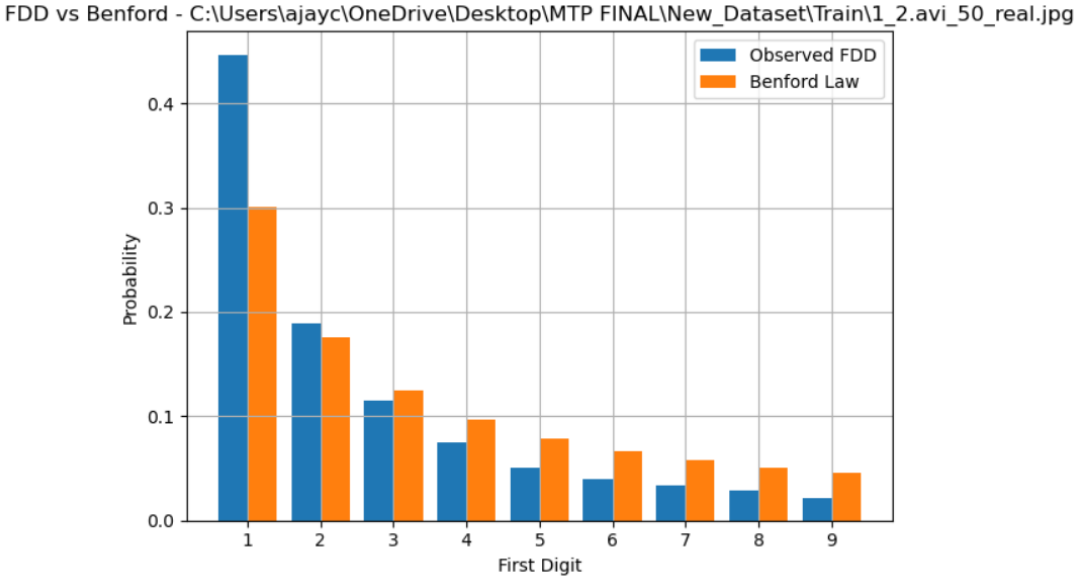

Figure 6.1: Histogram of DCT First Digit Distribution Values (Real)

```
Observed FDD: [0.45976116 0.18938214 0.10280374 0.07035306 0.05568536 0.04088785
 0.0336189  0.02557113 0.02193666]
KL Divergence from Benford: 0.0739600832111357
```

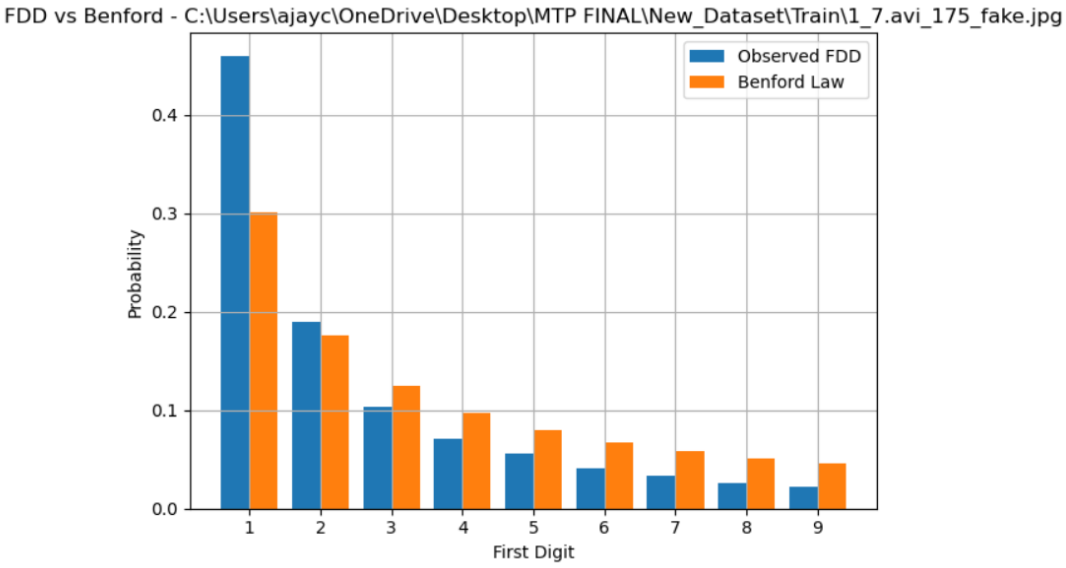

Figure 6.2: Histogram of DCT First Digit Distribution Values (Spoof)

6.3 Check with HOG Feature Vectors + Benford's law

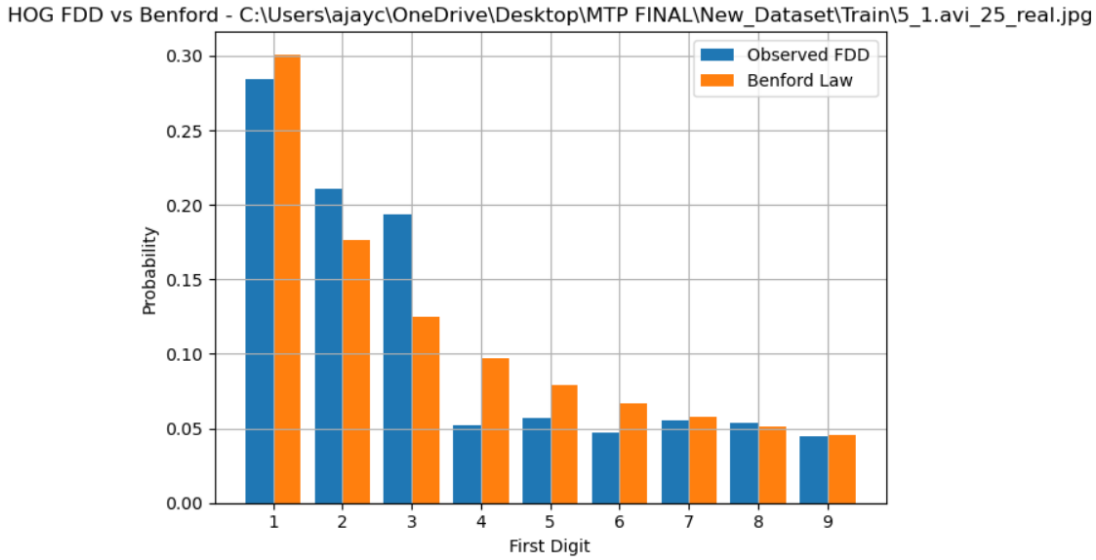

Figure 6.3: Histogram of HOG First Digit Distribution Values (Real)

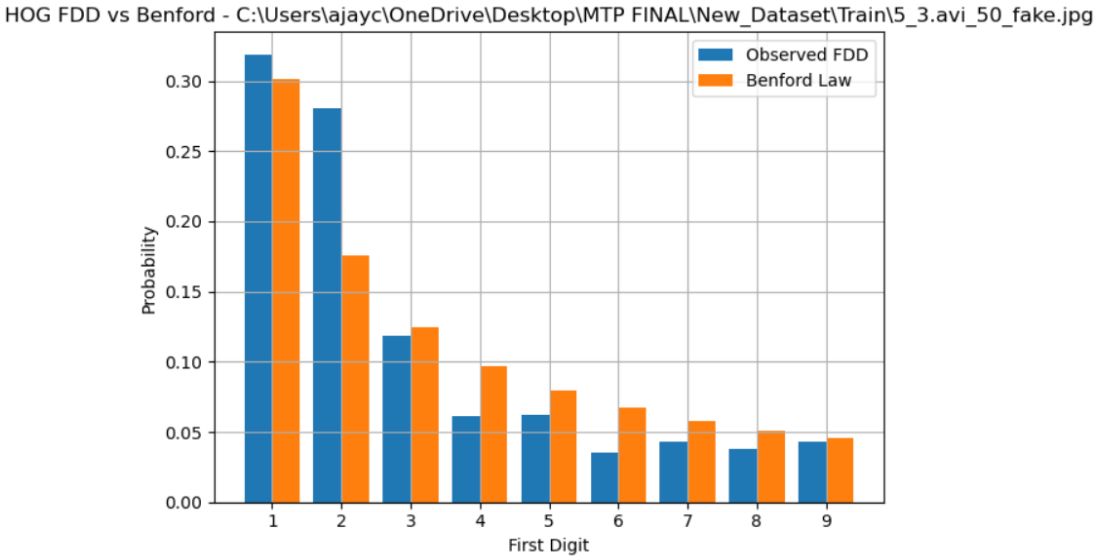

Figure 6.4: Histogram of HOG First Digit Distribution Values (Spoof)

6.4 CASIA SPATIAL OBSERVATION

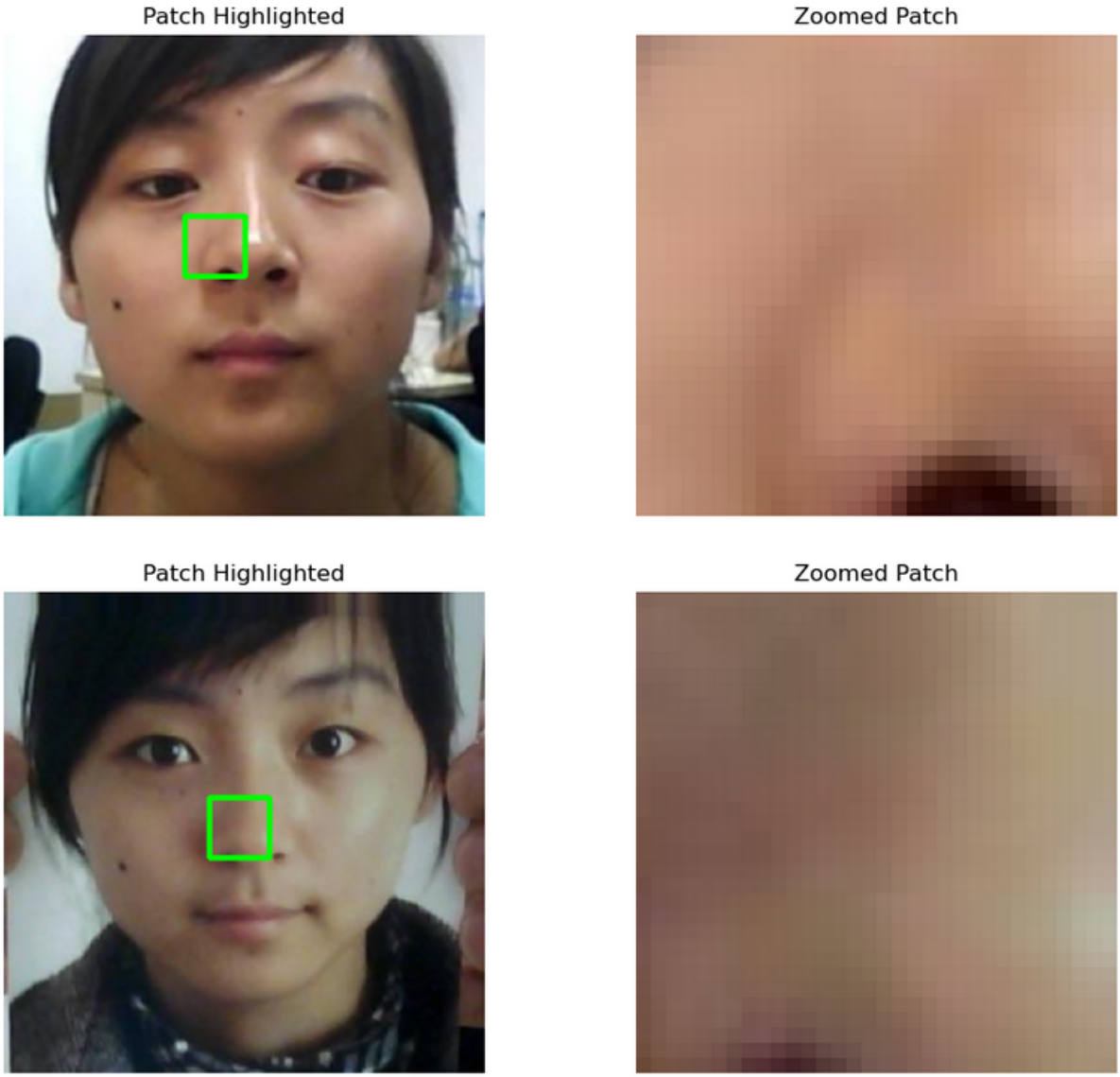

Figure 6.5: Visualization of SPATIAL between Original & Spoof image

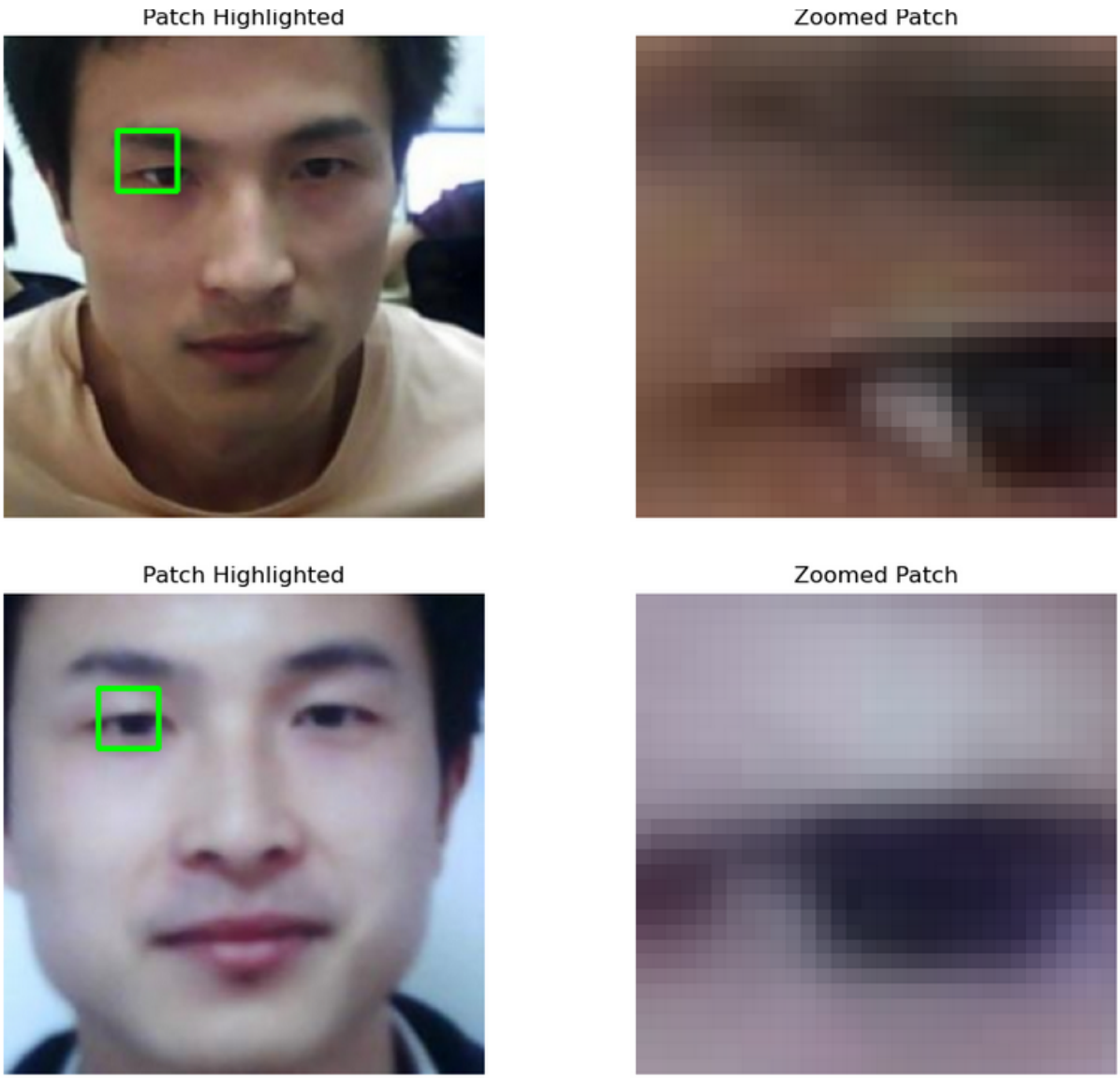

Figure 6.6: Visualization of SPATIAL between Original & Spoof image

6.4.1 Observations with SPATIAL Analysis on CASIA

1. Real Images Contain Natural Texture Variability :

- Original (digital) images retain sharp textures, shadows, and lighting variations.
- A Benford deviation score is computed using distance metrics such as Euclidean distance between the empirical and theoretical Benford distributions.

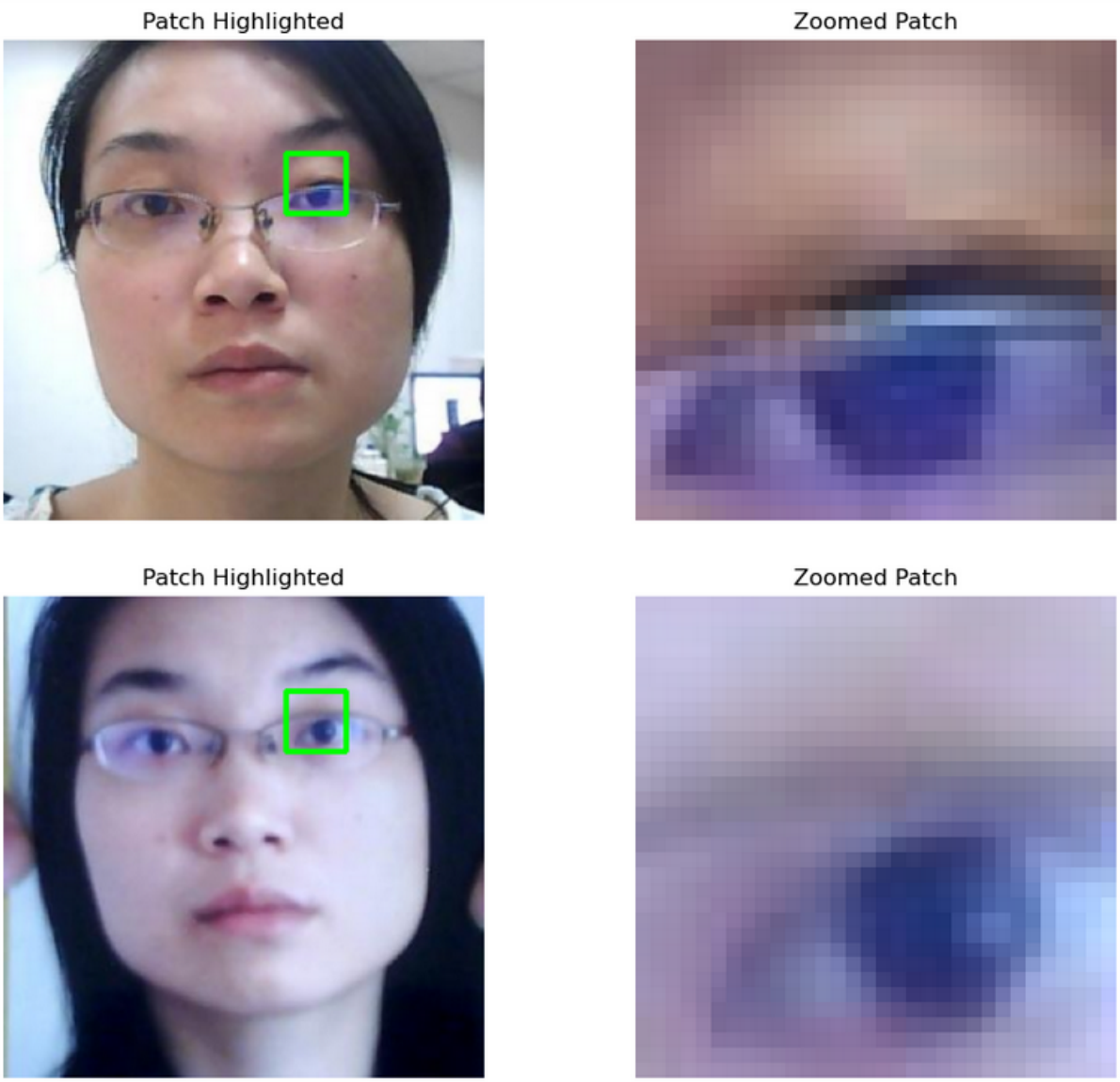

Figure 6.7: Visualization of SPATIAL between Original & Spoof image

2. Smoothing Artifacts in Printed Spoofs :

- Printed spoof images often undergo blurring or surface smoothing due to printer limitations or camera focus.
- This reduces local pixel variance, causing neighboring blocks to appear more similar, hence lower Euclidean distance.

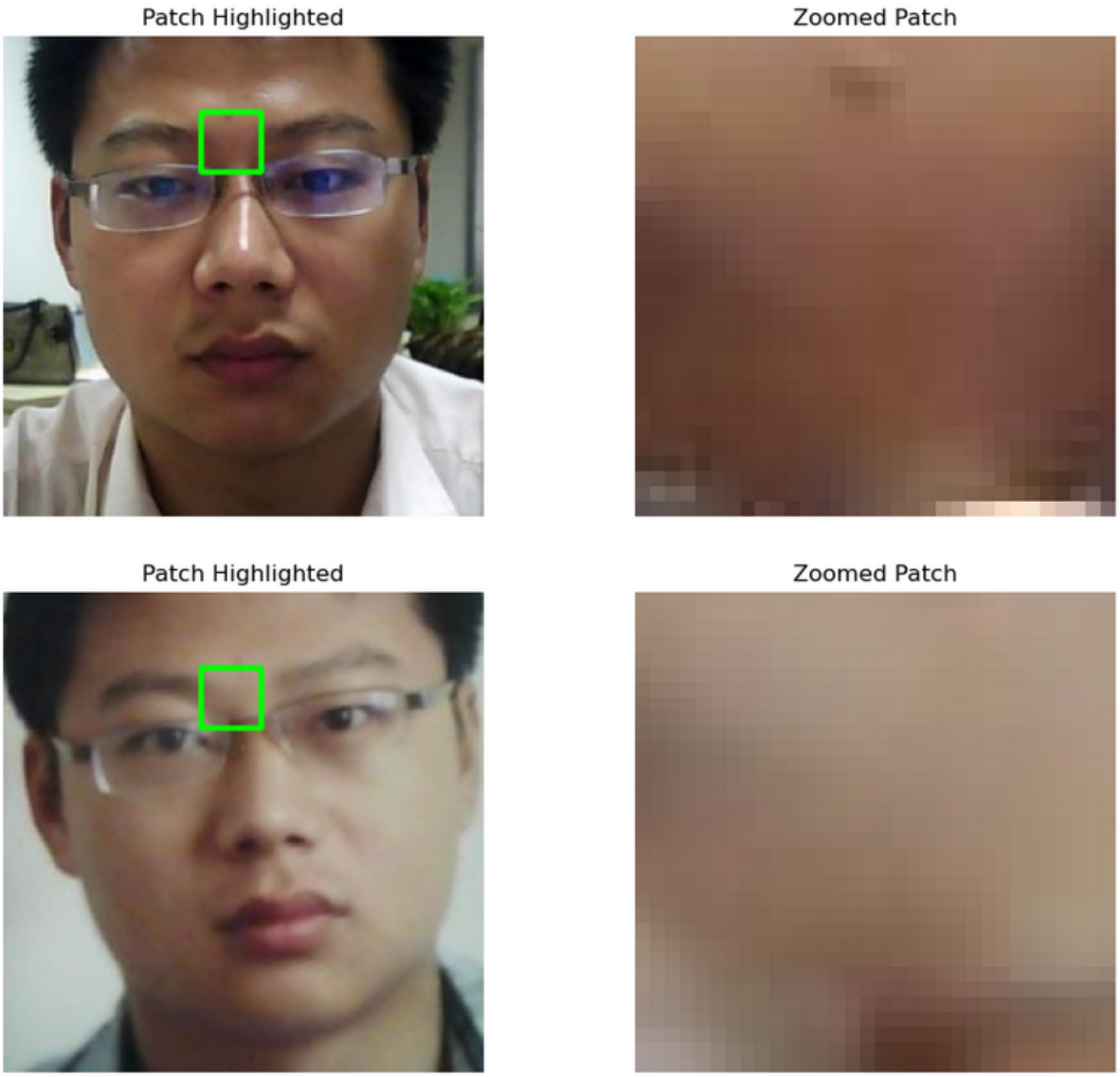

Figure 6.8: Visualization of SPATIAL between Original & Spoof image

Chapter 7

Conclusion

DCT vs HOG Features with Benford's Law

In this study, both Discrete Cosine Transform (DCT) and Histogram of Oriented Gradients (HOG) feature vectors were extracted from facial image patches and evaluated using Benford's Law through First Digit Distribution (FDD) analysis to distinguish between original and spoofed images.

Feature Type	**Purpose**	**Strength**
DCT	Captures frequency-domain information from image patches	Reveals compression artifacts and distinguishes smooth vs. textured areas.
HOG	Captures edge and gradient orientation histograms	Effective for analyzing shape, structure, and texture patterns.Useful for structural texture analysis but less sensitive to the digit distribution shifts that characterize spoofing artifacts.
Benford FDD	Checks for statistical deviations from natural digit distributions	Useful for detecting manipulations or unnatural patterns in feature distributions (e.g., spoof images)

Table 7.1: Summary of Feature types and their Uses in Face Anti-Spoofing

Bibliography

[1] H. Vinutha and G. Thippeswamy, "Antispoofing in face biometrics: A comprehensive study on software-based techniques," *Computer Science and Information Technologies*, vol. 4, no. 1, pp. 1–13, 2023.

[2] K. Karthik and B. R. Katika, "Face anti-spoofing based on sharpness profiles," in *2017 IEEE international conference on industrial and information systems (ICIIS)*, IEEE, 2017, pp. 1–6.

[3] I. Pavlidis and P. Symosek, "The imaging issue in an automatic face/disguise detection system," in *Proceedings IEEE Workshop on Computer Vision Beyond the Visible Spectrum: Methods and Applications (Cat. No. PR00640)*, IEEE, 2000, pp. 15–24.

[4] T. I. Dhamecha, R. Singh, M. Vatsa, and A. Kumar, "Recognizing disguised faces: Human and machine evaluation," *PloS one*, vol. 9, no. 7, e99212, 2014.

[5] T. Edmunds, "Protection of 2d face identification systems against spoofing attacks," Ph.D. dissertation, Université Grenoble Alpes, 2017.

[6] F. A. Pujol, M. J. Pujol, C. Rizo-Maestre, and M. Pujol, "Entropy-based face recognition and spoof detection for security applications," *Sustainability*, vol. 12, no. 1, p. 85, 2019.

[7] M. R. Hasan, S. H. Mahmud, and X. Y. Li, "Face anti-spoofing using texture-based techniques and filtering methods," in *Journal of Physics: Conference Series*, IOP Publishing, vol. 1229, 2019, p. 012 044.

[8] L. Feng, L.-M. Po, Y. Li, *et al.*, "Integration of image quality and motion cues for face anti-spoofing: A neural network approach," *Journal of Visual Communication and Image Representation*, vol. 38, pp. 451–460, 2016.

[9] K. Karthik and B. R. Katika, "Image quality assessment based outlier detection for face anti-spoofing," in *2017 2nd international conference on communication systems, computing and IT applications (CSCITA)*, IEEE, 2017, pp. 72–77.

[10] J. Yang, Z. Lei, and S. Z. Li, "Learn convolutional neural network for face anti-spoofing," in *2014 International Conference on Pattern Recognition (ICPR)*, IEEE, 2014, pp. 1237–1242.

[11] Y. Liu, A. Jourabloo, and X. Liu, "Learning deep models for face anti-spoofing: Binary or auxiliary supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 389–398.

[12] A. Jourabloo, Y. Liu, and X. Liu, "Face de-spoofing: Anti-spoofing via noise modeling," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 290–306.

[13] Anonymous, *Vitran: Transformer-based face anti-spoofing with vision-language pretraining*, Please replace with actual authors and venue if known, 2022.

[14] K. Kollreider, H. Fronthaler, and J. Bigun, "Non-intrusive liveness detection by face images," *Image and Vision Computing*, vol. 27, no. 3, pp. 233–244, 2009.