






Article

Pixelator v2: A Novel Perceptual Image Comparison Method with LAB Colour Space and Sobel Edge Detection for Enhanced Security Analysis

Somdip Dey ^{1,*} , Jabir Alshehabi Al-Ani ¹ , Aikaterini Bourazeri ² , Suman Saha ³, Rohit Purkait ³, Samuel Hill ¹  and Julian Thompson ¹ 

¹ Department of Data Science, York St. John University, York YO31 7EL, UK;

j.alshehabialani@yorks.j.ac.uk (J.A.A.-A.); s.hill@yorks.j.ac.uk (S.H.); j.thompson4@yorks.j.ac.uk (J.T.)

² School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, UK; a.bourazeri@essex.ac.uk

³ Nosh Technologies, Colchester CO4 3SL, UK; suman.saha@nosh.tech (S.S.); r.purkait@nosh.tech (R.P.)

* Correspondence: s.dey@yorks.j.ac.uk or dey@nosh.tech

Abstract: In this paper, we introduce Pixelator v2, a novel perceptual image comparison method designed to enhance security and analysis through improved image difference detection. Unlike traditional metrics such as MSE, Q, and SSIM, which often fail to capture subtle but critical changes in images, Pixelator v2 integrates the LAB (CIE-LAB) colour space for perceptual relevance and Sobel edge detection for structural integrity. By combining these techniques, Pixelator v2 offers a more robust and nuanced approach to identifying variations in images, even in cases of minor modifications. The LAB colour space ensures that the method aligns with human visual perception, making it particularly effective at detecting differences that are less visible in RGB space. Sobel edge detection, on the other hand, emphasises structural changes, allowing Pixelator v2 to focus on the most significant areas of an image. This combination makes Pixelator v2 ideal for applications in security, where image comparison plays a vital role in tasks like tamper detection, authentication, and analysis. We evaluate Pixelator v2 against other popular methods, demonstrating its superior performance in detecting both perceptual and structural differences. Our results indicate that Pixelator v2 not only provides more accurate image comparisons but also enhances security by making it more difficult for subtle alterations to go unnoticed. This paper contributes to the growing field of image-based security systems by offering a perceptually-driven, computationally efficient method for image comparison that can be readily applied in information system security.

Keywords: perceptual image comparison; CIE LAB colour space; sobel edge detection; image security; structural analysis; tamper detection; image processing algorithms; information visualisation



Citation: Dey, S.; Alshehabi Al-Ani, J.; Bourazeri, A.; Saha, S.; Purkait, R.; Hill, S.; Thompson, J. Pixelator v2: A Novel Perceptual Image Comparison Method with LAB Colour Space and Sobel Edge Detection for Enhanced Security Analysis. *Electronics* **2024**, *13*, 4541. <https://doi.org/10.3390/electronics13224541>

Academic Editor: Stefanos Kollias

Received: 25 September 2024

Revised: 13 November 2024

Accepted: 15 November 2024

Published: 19 November 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The need for robust image comparison methods has become increasingly important in a wide range of applications, such as image security [1–10], steganography [5,11–14], and watermarking [11,15–17]. These domains require accurate detection of image tampering or subtle modifications, which, if left undetected, could compromise the integrity and authenticity of images used in digital communication and forensic investigations. In particular, the ability to compare images to ensure that no unauthorised modifications have been made is crucial in ensuring tamper-proofing, content authenticity, and securing digital assets.

Several image comparison techniques have been proposed over the years, including Mean Squared Error (MSE) [18], the Quality of Index (Q) methodology or Q value [19], and the Structural Similarity Index Measure (SSIM) [20]. While these methods have been widely adopted, they exhibit significant limitations. MSE, for instance, focuses purely on pixel-wise intensity differences and does not account for perceptual or structural variations in images,

often yielding results that are misaligned with human visual perception. SSIM and Q value, although more perceptually relevant, can still fail to detect subtle, pixel-level changes in images that may be critical in scenarios such as steganography and watermarking, where every pixel matters. These limitations underscore the need for a more sensitive and comprehensive image comparison method.

To address these shortcomings, we propose **Pixelator v2**, an enhancement of the original **Pixelator** [21,22]. Pixelator v2 combines both pixel-wise RGB (red, green and blue) differences with perceptual analysis in the LAB colour space [23,24], thereby capturing both low-level and perceptually significant changes in images. Additionally, it incorporates Sobel edge detection [25–28] to emphasise structural changes, further enhancing its ability to detect tampering. This dual-layered approach ensures that Pixelator v2 can detect subtle modifications, such as those introduced in steganographic processes, which are often missed by conventional techniques.

In comparison to other edge detection techniques such as Canny [29], the Sobel edge detection [25–28] method used in Pixelator v2 is chosen for its balance between computational efficiency and structural sensitivity. While the Canny edge detector provides detailed edge detection, it is more sensitive to noise and computationally intensive due to steps like gradient computation and non-maximum suppression. By integrating the LAB colour space [23,24], Pixelator v2 achieves a closer alignment with human visual perception, enhancing subtle change detection critical for security applications. This combined approach offers a practical alternative to modern deep learning-based methods [30–32], which, despite their accuracy, require extensive computational resources and training data to generalize effectively across various domains.

The limitations of traditional methods are well-illustrated through a motivational case study comparing the **Lenna original image** [33] and a **distorted Lenna image**. Lenna is a standard test image used in the study of digital image processing since 1973. In this experiment (output is shown in Figure 1), the Green channel of RGB of the Lenna image was incrementally altered by just 1 pixel value, a subtle change that is imperceptible to the human eye. When comparing the two images, the **SSIM Map** yielded a similarity value of **0.9972**, indicating near-perfect similarity and failing to detect the pixel-level modification. However, **Pixelator v2** successfully identified the changes, producing a **Pixelator value** of **1054.0784**, thereby demonstrating its sensitivity to even the smallest alterations.

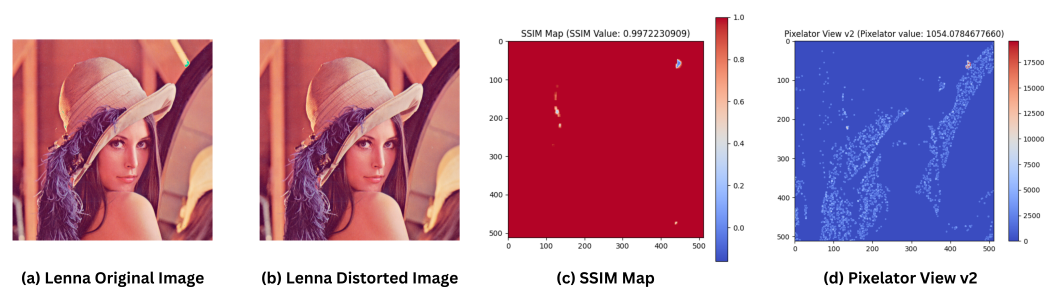


Figure 1. Highlighting differences between distorted Lenna and reference Lenna images using SSIM and Pixelator v2. (a) Is the unaltered original Lenna image; (b) is the altered Lenna image where the Green channel was incrementally altered by just 1 pixel value; (c) SSIM Map showing the differences between the original and the altered image; (d) Pixelator View v2 showing the differences between the original and the altered image.

Furthermore, Pixelator v2 provides additional insights by reporting several metrics alongside the **Pixelator value**, such as the **Pixelator RGB value**, **Pixelator LAB value**, **Total RGB Image Score**, **RGB Image Difference**, **Total LAB Image Score**, and **LAB Image Difference**. These additional metrics (as explained in Section 3.3.2) offer a more granular understanding of the differences between the images, making the tool not only more robust but also more informative for users analysing changes in security-sensitive images.

The contributions of this paper are as follows:

- We introduce **Pixelator v2**, a novel image comparison method that combines RGB and LAB colour space analysis with edge detection, significantly improving upon existing state-of-the-art techniques such as MSE, Q value, and SSIM.
- We provide a comprehensive set of metrics alongside the Pixelator value, enabling a deeper understanding of image differences for both perceptual and pixel-level changes.
- We demonstrate, through experimental results, that Pixelator v2 outperforms state-of-the-art approaches in detecting subtle image modifications, highlighting its potential for applications in image security, steganography, and tamper detection.

The rest of this paper is structured as follows: Section 2 discusses related works on image comparison methods, Section 3 details the design and implementation of Pixelator v2, Section 4 presents the experimental and validation results, and Section 5 discusses some of the relevant observations from experimental results and analysis, followed by a conclusion in Section 6. Source code of Pixelator v2 is provided in Section 7.

2. Related Works

Image comparison has been a long-standing challenge in computer vision, particularly in applications such as image security, steganography, and watermarking. Traditional methods such as **Mean Squared Error (MSE)**, **Q value**, and **Structural Similarity Index Measure (SSIM)** have been widely employed. **MSE** computes the pixel-wise differences between two images and averages them, providing a simplistic error metric that fails to account for perceptual relevance or structural information. Similarly, **Q value** focuses on pixel-wise differences but provides better results than MSE by incorporating factors like luminance and contrast, yet still falls short in detecting more intricate differences. **SSIM** improved upon these by incorporating structure, contrast, and luminance into its similarity measure, making it more aligned with human visual perception. However, as demonstrated in our case study, SSIM can miss small but significant changes at the pixel level, especially when modifications occur in non-perceptually significant regions like isolated channel values. Other noteworthy measures include the **Information Theoretical-Based Statistic Similarity Measure (ISSM)** [34], which offers different perspectives on image quality by leveraging statistical and information theory-based approaches. Additionally, the **Feature Similarity Index Measure (FSIM)** [35] attempts to improve perceptual relevance by focusing on low-level features in images. Despite this, FSIM, like SSIM, struggles to capture nuanced pixel-level variations essential in security applications where minute changes are critical. While effective in specific contexts, these measures may not address the subtle structural and perceptual differences targeted by Pixelator v2.

Recent advances have introduced the use of **machine learning** and **deep learning** algorithms to improve image comparison techniques. The **Learned Perceptual Image Patch Similarity (LPIPS)** [30], **Fréchet Inception Distance (FID)** [31], and **Deep Image Structure and Texture Similarity (DISTS)** [32] metrics have gained attention due to their ability to leverage deep neural networks to model image similarity. **LPIPS** compares image patches through the learned features of a neural network, achieving results that are closer to human perception. However, it is computationally intensive and relies heavily on pre-trained models, which may not generalise well across different domains. **FID**, widely used to evaluate the quality of generated images, particularly in Generative Adversarial Network (GAN)-based models [36–38], focuses on capturing the distributional difference between feature vectors of real and generated images. While useful for high-level generative tasks, **FID** is less effective for detecting pixel-level alterations, which are crucial in image security contexts. **DISTS**, a more recent technique, combines structural and textural similarity, improving upon traditional perceptual similarity measures, but it too suffers from computational overhead and struggles with pixel-level precision in cases of subtle image manipulations.

Moreover it should be kept in mind that methods such as Convolutional Neural Networks (CNN) [39–43] and GAN-based models have demonstrated success in detecting complex patterns and features in images. However, their computational demands and

need for large training datasets limit their scalability and accessibility for real-time or resource-limited applications. Pixelator v2's design provides a balance by employing computationally efficient yet perceptually aligned methods suitable for security applications without the overhead of deep learning.

Recent advancements in pixel-level manipulation detection have provided new frameworks for identifying tampered or modified image regions with greater accuracy. Kong et al. [44] introduced a method called Pixel-inconsistency modeling for image manipulation localization, which identifies discrepancies in images by analyzing pixel-level inconsistencies. This technique enhances detection capabilities, especially in applications requiring precise localization of altered regions, thus aligning with the goals of Pixelator v2 to improve perceptual and structural difference detection in security-sensitive contexts.

Additionally, Prashnani et al. [45] proposed PieAPP (Perceptual Image-Error Assessment through Pairwise Preference), a perceptual image-error assessment metric that leverages human pairwise preference data. PieAPP estimates perceptual differences between images in a way that is closely aligned with human visual judgement, offering a benchmark for evaluating perceptual similarity and error. Pixelator v2's design similarly emphasizes perceptual sensitivity, specifically by combining LAB colour space and structural edge detection to capture subtle variations that might otherwise go unnoticed.

To provide a clearer comparison of the capabilities of existing popular methods against Pixelator v2, Table 1 summarizes each technique's approach, perceptual sensitivity, structural analysis, and computational efficiency. This comparison highlights the limitations of traditional and deep learning-based approaches, underscoring the need for a balanced solution that Pixelator v2 aims to achieve.

Table 1. Comparison of Some of the Existing Popular Techniques with Pixelator v2.

Method	Approach	Perceptual Sensitivity	Structural Analysis	Computational Efficiency
MSE [18]	Pixel-wise intensity	Low	None	High
SSIM [20]	Structural similarity	Moderate	Limited	Moderate
Q [19]	Gradient-based	Moderate	Moderate	Moderate
FSIM [35]	Feature-based similarity	High	Moderate	Low
DISTS [32]	Deep feature similarity	High	Moderate	Moderate
PieAPP [45]	Pairwise perceptual assessment	High	None	Low
LPIPS [30]	Deep learning	High	None	Low
FID [31]	GAN-based	High	None	Low
Pixelator v2 (proposed)	LAB + Sobel	High	High	High

Pixelator v2 addresses the aforementioned limitations by introducing a dual approach that combines both **pixel-wise RGB differences** and **perceptual analysis in the LAB colour space**. This method not only captures fine-grained pixel changes but also provides perceptually relevant insights into image differences. The inclusion of **Sobel edge detection** further enhances Pixelator v2's ability to detect structural changes, enabling it to perform robustly in applications where image tampering, steganography, and watermarking are critical concerns. Compared to traditional methods and deep learning-based approaches, Pixelator v2 strikes a balance between computational efficiency and perceptual accuracy, making it a powerful tool for a wide range of image comparison tasks.

As shown in Table 1, while traditional methods such as MSE, Q, and SSIM provide relatively high computational efficiency but struggle with detecting subtle changes. More advanced deep learning techniques such as LPIPS, FID, and DISTS, though effective in perceptual analysis, are often computationally expensive or domain-specific. **Pixelator v2** overcomes these challenges by integrating pixel-wise and perceptual analysis with structural detection. This combination provides a comprehensive image comparison frame-

work/tool suitable for high-stakes applications such as image security, tamper detection, and forensic analysis.

3. Pixelator v2 Methodology

3.1. Overview of Pixelator v2

Pixelator v2 is an advanced image comparison tool designed to overcome the limitations of existing methods like MSE, Q, and SSIM by integrating pixel-wise and perceptual analysis. It leverages both the RGB colour space for low-level pixel differences and the CIE-LAB colour space for perceptual relevance. Moreover, it incorporates Sobel edge detection to highlight structural changes in the images, providing a comprehensive method for detecting subtle modifications. Pixelator v2 offers additional metrics such as Pixelator RGB value, Pixelator LAB value, and structural differences to present a full-fledged comparative analysis of images.

3.2. Detailed Approach and Design Choices

Pixelator v2 is designed with a dual-layer approach to address both pixel-level and perceptual differences. The algorithm of Pixelator v2 is provided in Section 3.3.1. The key steps of the methodology include:

- **Pixel-wise RGB Difference Calculation:** Pixelator v2 computes the pixel-wise differences between two images in the RGB space as proposed in [21]. This captures minute pixel-level changes which might be important in image tampering detection.
- **LAB Colour Space for Perceptual Analysis:** The CIE-LAB colour space [23,24] is a perceptual colour model that represents colour in a manner aligned with human vision, aiming to approximate how humans perceive colour differences. Unlike the RGB colour model, which is device-dependent and primarily used for digital displays, CIE-LAB is device-independent, making it suitable for applications involving perceptual analysis. The use of the CIE-LAB colour space allows Pixelator v2 to capture perceptual differences that align with human vision. This is crucial for detecting changes that are less perceptible in the RGB space but noticeable to the human eye.
- **Sobel Edge Detection:** Sobel filtering is a widely used edge detection method in image processing [25–28]. The Sobel operator is a discrete differentiation operator that computes an approximation of the gradient of the image intensity function. This is done by convolving the image with two 3×3 kernels that are designed to detect vertical and horizontal edges, respectively. The Sobel filter highlights areas of an image where there is a significant intensity change, making it particularly useful for detecting edges and structural changes within an image. This is critical for applications such as image comparison, where edge information can reveal alterations like tampering or watermarking. Moreover, Sobel filtering is computationally efficient, which makes it advantageous for real-time applications. In our proposed Pixelator v2, to ensure structural changes are captured, Sobel filters are applied to the difference matrices, which enhance edge detection and structural modifications in the images. The decision to employ Sobel edge detection in Pixelator v2 stems from its balance between computational efficiency and sensitivity to structural changes. Sobel filtering is computationally lighter than alternatives such as the Canny edge detector [29], which involves additional steps such as gradient magnitude computation, non-maximum suppression, and hysteresis thresholding. While Canny is highly effective for detecting detailed edges, it is often more sensitive to noise and can produce more complex results, which may not be necessary for our intended application. Sobel's simpler approach, which uses convolution with small, integer-valued kernels to detect horizontal and vertical edges, allows for real-time application without sacrificing accuracy in identifying significant structural modifications. Moreover, Sobel's robustness in capturing primary edge information makes it a suitable choice for image comparison tasks where subtle tampering or structural changes need to be highlighted efficiently, as demonstrated in Pixelator v2's ability to detect pixel-level alterations in security-sensitive images.

After the aforementioned key steps are executed, Pixelator v2 shows the output (difference in the images) as **Pixelator View v2** (as demonstrated in Figure 2) along with the Pixelator value (up to 10 decimal points) as calculated from our heuristic based metrics (as mentioned in Section 3.3.2). Figure 2 shows the visual output (Pixelator View v2) of Pixelator v2 comparing the Lenna original image and a distorted Lenna image, where the Green channel of RGB of the Lenna image was incrementally altered by just 1 pixel value. In Figure 2, it can be noticed that Pixelator View v2 highlights the changes/differences in pixel along the dimensions of the image, which is 512×512 for the Lenna image.

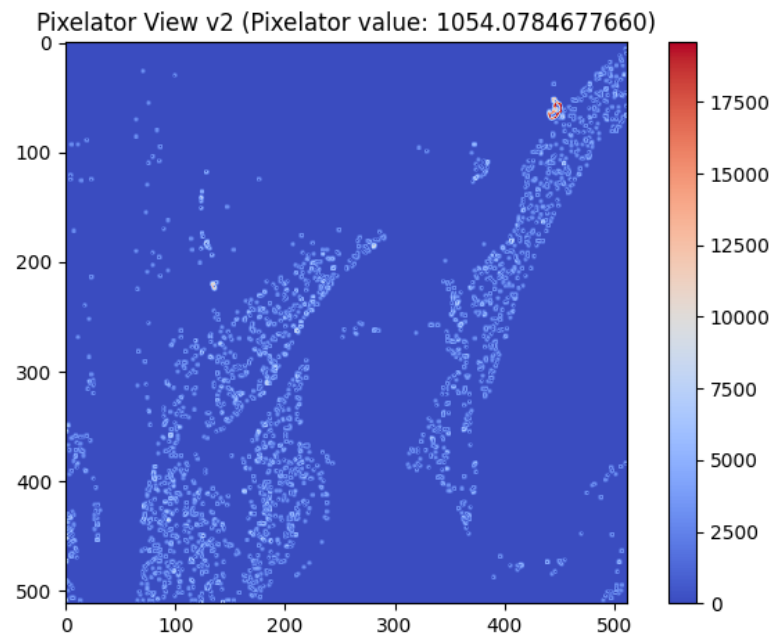


Figure 2. Highlighting differences between distorted Lenna image (where the Green channel of RGB of the Lenna image was incrementally altered by just 1 pixel value) and original Lenna image using **Pixelator View v2** along with Pixelator value.

3.3. Handling Greyscale Images in Pixelator v2

Pixelator v2 is also capable of comparing differences in greyscale images. When comparing greyscale images, Pixelator v2 processes them in a manner similar to colour images, but with certain considerations for the single intensity channel that greyscale images possess. Unlike RGB images, which have three separate (RGB) channels for red, green, and blue, greyscale images consist of a single channel representing intensity values. In Pixelator v2, the algorithm first treats this single intensity channel as analogous to the luminosity component in the CIE-LAB colour space. Thus, the perceptual differences are effectively captured even in the absence of chromatic information, aligning with how human vision perceives contrast and brightness changes in greyscale imagery.

However, it is important to note that while Pixelator v2 remains effective in identifying pixel-level alterations and structural differences using Sobel filtering in greyscale images, the absence of chromatic data may limit the tool's ability to capture certain perceptual nuances that arise from colour variations in RGB images. This could potentially reduce its sensitivity in detecting subtle perceptual changes that depend on chromatic contrast rather than intensity alone. Nevertheless, the incorporation of Sobel edge detection compensates for this limitation to some extent by enhancing the detection of structural changes, making Pixelator v2 sufficiently robust for greyscale image comparison tasks.

To ensure consistency and reproducibility of the proposed work, we confirm that the image intensity range for both RGB and greyscale images used in this study is within the standard range of 0–255 for each channel. This range was selected to align with common practices in digital image processing, where pixel values typically span from 0 (representing

black) to 255 (representing white in greyscale or the maximum intensity for each color channel in RGB).

3.3.1. Algorithm for Pixelator v2

The following algorithm (Algorithm 1) outlines the steps involved in the Pixelator v2 process:

Algorithm 1 Pixelator v2: Image Comparison Algorithm

- 1: **Input:** Image I_s (Source Image), Image I_b (Baseline Image)
 - 2: Convert I_s and I_b to RGB
 - 3: Calculate pixel-wise differences in RGB space
 - 4: Compute pixelator value in RGB (*Pixelator RGB Value*) as shown in Equation (1) in Section 3.3.2
 - 5: Convert I_s and I_b to LAB Colour Space
 - 6: Calculate perceptual differences in LAB space (*Pixelator LAB Value*) as shown in Equation (2) in Section 3.3.2
 - 7: Combine RGB and LAB values (called as *Pixelator Value*) as shown in Equation (3) in Section 3.3.2
 - 8: Apply Sobel filters to the difference matrices (i.e., the pixel-wise magnitude of difference between I_s and I_b in both RGB and LAB spaces) to enhance edge detection and structural comparison.
 - 9: **Output:** Combined Pixelator value, Sobel-filtered image
-

3.3.2. Metrics and Formulas

Pixelator v2 introduces multiple metrics to provide a comprehensive analysis:

- **Pixelator RGB Value:** This metric measures pixel-wise differences in the RGB space, providing a low-level comparison of the two images. The Pixelator RGB value is calculated based on the pixel-wise difference between two images in the RGB colour space. The formula is:

$$\text{Pixelator RGB Value} = \frac{\sum_{i=1}^n |I_s(i) - I_b(i)|}{\text{Width} \times \text{Height}} \quad (1)$$

where:

- $I_s(i)$ is the pixel intensity of the source image at index i .
- $I_b(i)$ is the pixel intensity of the baseline image at index i .
- Width and Height are the dimensions of the image.
- **Pixelator LAB Value:** This metric measures perceptual differences in the LAB colour space, aligning with human visual perception. The Pixelator LAB value is calculated using the perceptual difference between two images in the LAB colour space. The formula is:

$$\text{Pixelator LAB Value} = \frac{\sum_{i=1}^n \|I_s^{LAB}(i) - I_b^{LAB}(i)\|}{\text{Width} \times \text{Height}} \quad (2)$$

where:

- $I_s^{LAB}(i)$ and $I_b^{LAB}(i)$ represent the pixel values of the source and baseline images in the LAB colour space.
- $\|\cdot\|$ represents the Euclidean distance between the pixel values in the LAB colour space.

It should be noted that the choice of L1 norm for the Pixelator RGB value in Equation (1) and L2 norm for the Pixelator LAB value in Equation (2) was made based on the distinct characteristics of these colour spaces and the types of differences they are designed to capture. The L1 norm, also known as the Manhattan distance [46,47], is particularly effective for capturing pixel-wise differences in the RGB space where the focus is on intensity changes

between individual pixel values. This norm emphasizes pixel-level differences by directly summing the absolute differences, which is appropriate in the RGB space where each channel represents a separate intensity level (red, green, or blue), making L1 a natural fit.

In contrast, the LAB colour space is designed to model human visual perception more closely, capturing differences in luminance and chromatic components (L, a, and b channels). In this context, the L2 norm, or Euclidean distance [46–48], is better suited to measure perceptual differences since it accounts for the overall magnitude of changes in both intensity and colour perception. By squaring the differences, the L2 norm gives greater weight to larger deviations, making it more sensitive to perceptual shifts that may be less noticeable in the RGB space but critical in LAB. Thus, the use of L2 norm in the LAB space aligns with the goal of capturing perceptual discrepancies more comprehensively.

To summarise, the Pixelator RGB Value is computed using the Manhattan (L1) norm, which directly sums the absolute pixel-wise differences between the source and baseline images in RGB space, as shown in Equation (1). This approach emphasizes precise pixel-level differences. For the LAB colour space, the Euclidean (L2) norm is used, as shown in Equation (2), which captures perceptual variations by measuring the overall magnitude of changes in luminance and chromaticity. The combined Pixelator Value, shown in Equation (3), reflects the aggregate perceptual and structural differences between images.

- **Pixelator Value:** The combined Pixelator value from both RGB and LAB space comparisons. The formula is:

$$\text{Pixelator Value} = \text{Pixelator RGB Value} + \text{Pixelator LAB Value} \quad (3)$$

- **Total RGB Image Score:** The total score of pixel intensity differences in the RGB space is calculated as:

$$\text{Total RGB Score} = \frac{\sum_{i=1}^n |I_s(i)|}{\text{Width} + \text{Height}} \quad (4)$$

where:

- $I_s(i)$: Pixel intensity of the source image at index i .
- n : The total number of pixels in the image.
- Width and Height : The dimensions of the image.

- **Total LAB Image Score:** The total perceptual score in the LAB space is:

$$\text{Total LAB Score} = \frac{\sum_{i=1}^n \|I_s^{LAB}(i)\|}{\text{Width} + \text{Height}} \quad (5)$$

where:

- $I_s^{LAB}(i)$: Pixel value of the source image in the LAB colour space at index i .
- n : The total number of pixels in the image.
- Width and Height : The dimensions of the image.

- **Total Image Score:** The total score for an image, either in RGB or LAB space, is calculated by summing the pixel intensities across all pixels. The formula is:

$$\text{Total Image Score} = \frac{\sum_{i=1}^n I_s(i)}{\text{Width} + \text{Height}} \quad (6)$$

where:

- $I_s(i)$ is the pixel intensity of the source image at index i (in RGB or LAB space).
- n is the total number of pixels in the image.
- Width and Height are the dimensions of the image.

- **Image Difference:** The difference between the two images in both RGB and LAB spaces is represented as a percentage, calculated as:

$$\text{Image Difference} = 100.0 - \left(\frac{\text{Pixelator Value}}{\text{Total Image Score}} \times 100 \right) \quad (7)$$

where:

- *Pixelator Value:* The combined Pixelator value from both RGB and LAB space comparisons as shown in Equation (3).
- *Total Image Score:* The total image score calculated either in RGB or LAB space as shown in Equation (6).

These aforementioned metrics, alongside the combined Pixelator value, offer a detailed and versatile tool for image comparison, making Pixelator v2 an advanced solution for tamper detection, security analysis, and steganography.

3.3.3. Justification of Metrics and Design Choices

The metrics introduced in Pixelator v2 provide substantial improvements over traditional and recent image comparison techniques in the following ways:

- **Pixelator RGB and LAB Values:** Unlike MSE, which purely computes pixel-wise intensity differences, Pixelator v2 combines both RGB and LAB colour space information. The LAB colour space accounts for perceptual relevance, capturing differences that MSE misses due to its lack of perceptual modelling. This dual approach improves the precision of detecting subtle, imperceptible changes, which are especially relevant for security and tamper detection tasks.
- **Total Image Scores:** By calculating total scores in both RGB and LAB spaces, Pixelator v2 provides a holistic view of image differences, unlike SSIM, which focuses primarily on luminance, contrast, and structure but may miss pixel-level alterations. This added layer of analysis ensures that both perceptual and structural changes are accounted for, leading to more reliable image comparisons.
- **Sobel Edge Detection:** Sobel filtering highlights structural differences more effectively than methods such as LPIPS, FID, and DISTS, which often focus on learned perceptual similarity. Sobel-based structural analysis allows Pixelator v2 to detect changes in image edges and structures more precisely, which is crucial for applications like steganography and watermarking.
- **Pixelator Value:** The combined Pixelator value offers a comprehensive metric that captures both pixel-level and perceptual differences. This outperforms FID and DISTS, which, despite being learned measures, may fail to detect fine pixel-level changes due to the reliance on deep feature embeddings. Pixelator v2 bridges this gap by ensuring pixel-precision as well as perceptual insight.
- **Image Difference as Percentage:** The percentage-based image difference metric enables an intuitive understanding of how much the images differ, providing a clearer interpretation compared to Q or SSIM, which output similarity scores without a concrete representation of the extent of differences.

These improvements ensure that Pixelator v2 offers a robust, accurate, and interpretable image comparison methodology that excels in various domains including image security, tamper detection, and perceptual analysis.

4. Experimental and Validation Results

4.1. Database and Images Used

The experiments conducted in this research used images from the publicly available USC-SIPI Image Database [49]. This database is widely recognised in the field of image processing and analysis, offering a variety of image types such as greyscale, colour, and infra-red images. For this paper, we selected three representative images from this database:

2.1.02 (Colour Image of San Diego, 512×512), 3.2.25 (Grey Image of Pentagon, 1024×1024), and wash-ir (Infra-red image of Washington, D.C.) from “Volume 2: Aerials” (as shown in Figure 3). These images provide diversity in terms of dimension and colour scale, allowing for a comprehensive evaluation of Pixelator v2. Additionally, it should also be kept in mind that the images from the USC-SIPI Image Database are in TIFF (Tagged Image File Format), which is typically used for high-quality graphics.

In addition to the USC-SIPI Image Database, we have also utilized the KADID-10k Dataset [50] to evaluate the output of the Pixelator v2 methodology. The KADID-10k database contains an “image” folder with 81 reference images and 10,125 distorted images (resulting from 81 reference images \times 25 types of distortion \times 5 levels of distortion). Each image is saved in PNG format. The naming convention for distorted images follows the format *lxx_yy_zz.png*, where *xx* denotes the image ID, *yy* signifies the type of distortion, and *zz* represents the level of distortion; for example 01 would be lowest level of distortion, whereas, 05 will represent the highest level of distortion.



(a) 2.1.02 (Colour Image of San Diego, 512×512)

(b) 3.2.25 (Grey Image of Pentagon, 1024×1024)

(c) wash-ir (Infra-red image of Washington, D.C.)

Figure 3. Representative images for experiments consisting of (a) 2.1.02 (Colour Image of San Diego, 512×512); (b) 3.2.25 (Grey Image of Pentagon, 1024×1024); and (c) wash-ir (Infra-red image of Washington, D.C.) from “Volume 2: Aerials” of the USC-SIPI Image Database [49].

The KADID-10k dataset includes various types of distortions across several categories, which serve to thoroughly assess the sensitivity and accuracy of Pixelator v2 in capturing perceptual and structural differences. The distortions are summarized as follows:

- **Blurs:** Includes Gaussian blur (Distortion Type 01), lens blur (Distortion Type 02), and motion blur (Distortion Type 03).
- **Colour Distortions:** Encompasses colour diffusion (Distortion Type 04), colour shift (Distortion Type 5), colour quantization (Distortion Type 06), and two types of colour saturation adjustments (Distortion Type 07 and 08 respectively).
- **Compression:** JPEG2000 (Distortion Type 09) and JPEG (Distortion Type 10) compression.
- **Noise:** White noise (Distortion Type 11), white noise in colour components (Distortion Type 12), impulse noise (Distortion Type 13), multiplicative noise (Distortion Type 14), and denoise techniques (Distortion Type 15).
- **Brightness Change:** Adjustments to brightness (Distortion Type 16), darkening (Distortion Type 17), and mean shift (Distortion Type 18).
- **Spatial Distortions:** Includes jitter (Distortion Type 19), non-eccentricity patch (Distortion Type 20), pixelation (Distortion Type 21), quantization (Distortion Type 22), and random colour block insertion (Distortion Type 23).
- **Sharpness and Contrast:** High sharpen (Distortion Type 24) and contrast change distortions (Distortion Type 25).

By incorporating these varied distortions, the KADID-10k dataset offers a comprehensive benchmark to test Pixelator v2’s effectiveness in detecting subtle changes across a broad spectrum of image distortions, particularly for applications in security and perceptual image analysis.

The choice of the USC-SIPI and KADID-10k datasets reflects a need for diverse image types and varying levels of degradation. The USC-SIPI dataset includes high-quality

colour and greyscale images, while the KADID-10k dataset provides a spectrum of distortions (e.g., noise, compression, and colour changes), which are commonly encountered in security-sensitive contexts. By testing on these datasets, Pixelator v2 demonstrates robustness against both minor and pronounced visual alterations, mimicking real-world scenarios of image tampering, steganography, and watermarking.

4.2. Experiment Set 1: One-Pixel Alteration in the R Channel

In the first experiment, one random pixel in the red (R) channel of RGB was incrementally altered by a value of 1 in each of the selected images (from Figure 3). This experiment aimed to demonstrate Pixelator v2’s sensitivity to subtle, pixel-level changes that might be overlooked by traditional image comparison techniques. Table 2 summarises the results for Pixelator v2 when comparing the original and altered images and Figure 4 shows the visual output of Pixelator v2 (Pixelator View v2) with respective Pixelator value.

Table 2. Pixelator v2 results for one-pixel alteration in R channel.

Image	Pixelator RGB Value	Pixelator LAB Value	Pixelator Value	Total Image Score	Image Difference
2.1.02 (San Diego)	0.0009765625	0.0	0.0009765625	30,541.015625	0.0000031975
3.2.25 (Pentagon)	0.00048828125	0.0	0.00048828125	67,593.99462890625	0.0000007224
wash-ir (Washington, D.C.)	0.0002222222222222223	0.0	0.0002222222222222223	142,055.51133333333	0.0000001564

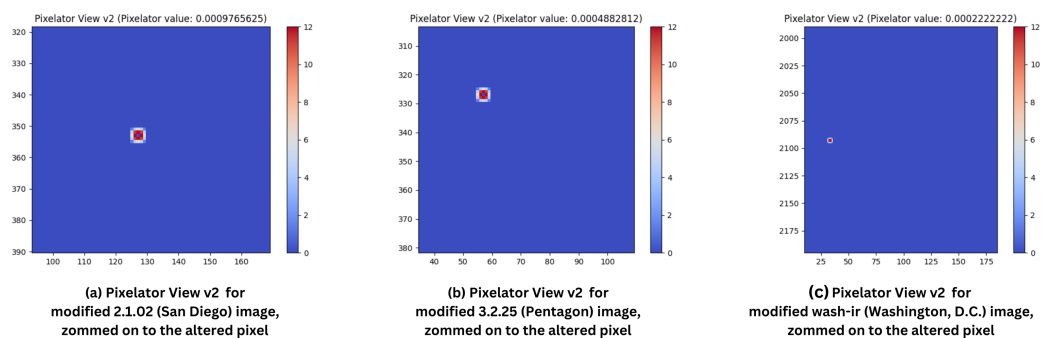


Figure 4. Zoomed in Pixelator View v2 for respective images for one-pixel alteration in R channel: (a) 2.1.02 (Colour Image of San Diego, 512 × 512); (b) 3.2.25 (Grey Image of Pentagon, 1024 × 1024); and (c) wash-ir (Infra-red image of Washington, D.C.) from “Volume 2: Aerials” of the USC-SIPI Image Database [49].

4.3. Experiment Set 2: Pixel Alteration in the Blue Channel

In this experiment, the blue (B) channel of the RGB images was altered by incrementally changing the value by 1. The aim was to evaluate whether Pixelator v2 could detect changes in a different channel than R (red) as Figure 2 and Figure 4 have already proved the efficacy of Pixelator v2 in detecting changes in pixel values in G and R channels respectively. Table 3 presents the Pixelator v2 results for this alteration and Figure 5 shows the visual output of Pixelator v2 (Pixelator View v2) with respective Pixelator value.

Table 3. Pixelator v2 results for pixel alteration in B channel.

Image	Pixelator RGB Value	Pixelator LAB Value	Pixelator Value	Total Image Score	Image Difference
2.1.02 (San Diego)	256.0	33,348.21133220212	33,604.21133220212	30,541.015625	0.838217049306138
3.2.25 (Pentagon)	512.0	81,358.0463598267	81,870.0463598267	67,593.99462890625	0.7574637403971991
wash-ir (Washington, D.C.)	1125.0	141,657.77327049721	142,782.77327049721	142,055.51133333333	0.7919439305386646

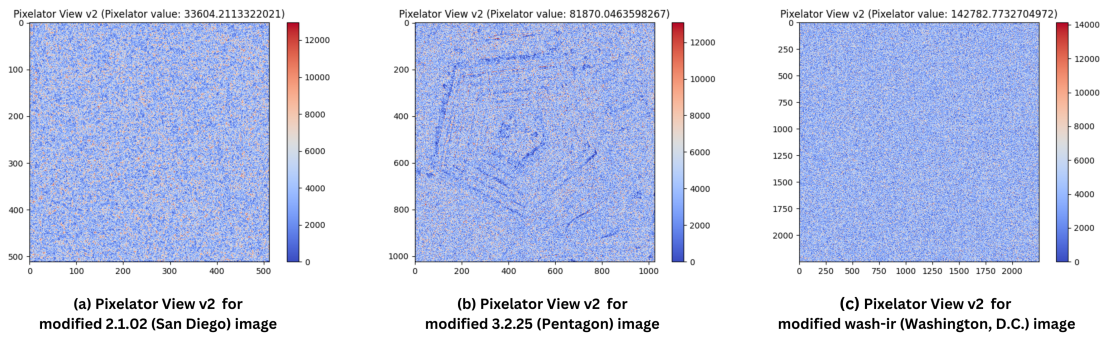


Figure 5. Pixelator View v2 for respective images for pixel alteration in B channel: (a) 2.1.02 (Colour Image of San Diego, 512×512); (b) 3.2.25 (Grey Image of Pentagon, 1024×1024); and (c) wash-ir (Infra-red image of Washington, D.C.) from “Volume 2: Aerials” of the USC-SIPI Image Database [49].

4.4. Experiment Set 3: Hidden Message Embedding

To test Pixelator v2’s performance in detecting steganographic alterations, the message “Hello, World!” was hidden in the selected images using a basic least significant bit (LSB) technique [51,52]. Table 4 shows the results for Pixelator v2 when comparing the original and altered images with the embedded message and and Figure 6 shows the visual output of Pixelator v2 (Pixelator View v2) with respective Pixelator value.

Table 4. Pixelator v2 results for hidden message embedding.

Image	Pixelator RGB Value	Pixelator LAB Value	Pixelator Value	Total Image Score	Image Difference
2.1.02 (San Diego)	2.9921875	5.109964529761771	8.102152029761772	30,541.015625	0.009797275692264407
3.2.25 (Pentagon)	2.24267578125	2.048850837438776	4.291526618688776	67,593.99462890625	0.0033178624722012273
wash-ir (Washington, D.C.)	0.014	0.6833622081875113	0.6973622081875113	142,055.51133333333	0.0000098553

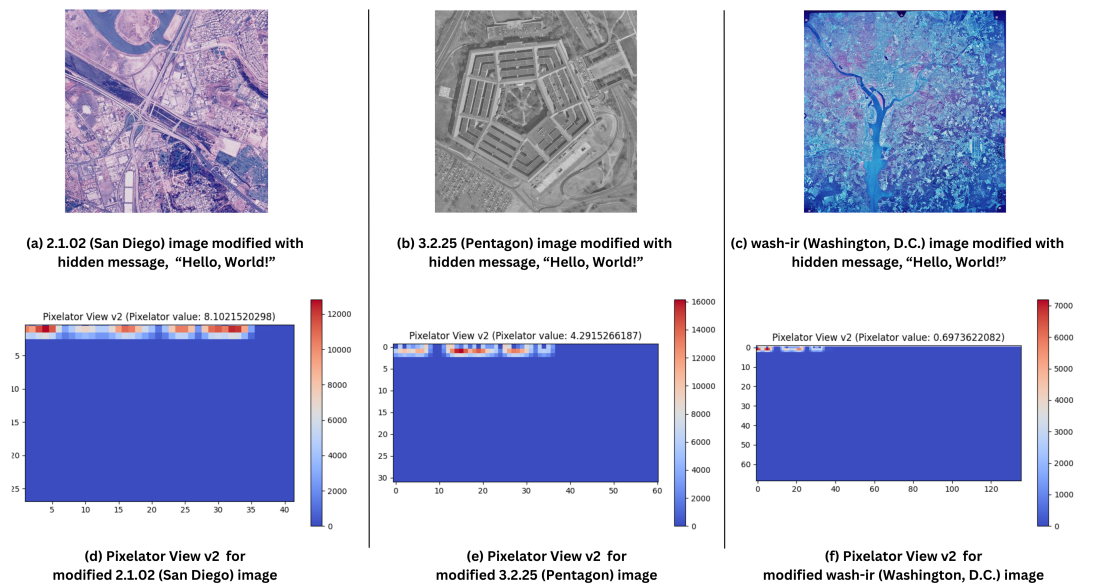


Figure 6. Pixelator View v2 for respective images with hidden message “Hello, World!” using LSB technique. (a) 2.1.02 (Colour Image of San Diego, 512×512) with hidden message; (b) 3.2.25 (Grey Image of Pentagon, 1024×1024) with hidden message; (c) wash-ir (Infra-red image of Washington, D.C.) with hidden message; (d) Pixelator View v2 of 2.1.02 image with hidden message; (e) Pixelator View v2 of 3.2.25 image with hidden message; and (f) Pixelator View v2 of wash-ir image with hidden message.

4.5. Experiment Set 4: Image Compression

This experiment evaluated Pixelator v2’s ability to detect changes in images after they were compressed. Compression is widely used in platforms like WhatsApp [53,54] and Instagram [55–57] for transmitting multimedia content. Given the fact that the images from the USC-SIPI Image Database are in TIFF (Tagged Image File Format) [58], which supports both lossy and lossless compression. Compression in TIFF images can be achieved using different methods, such as LZW (Lempel-Ziv-Welch) compression [59,60], which is lossless, and JPEG compression, which can be lossy. Given the fact that social media apps such as WhatsApp and Instagram alike use lossy compression like JPEG, each image was compressed using the standard JPEG algorithm (quality at 85%), and the results are shown in Table 5. Figure 7 shows the visual output of Pixelator v2 (Pixelator View v2) with respective Pixelator value for this set of experiments.

Table 5. Pixelator v2 results for image compression.

Image	Pixelator RGB Value	Pixelator LAB Value	Pixelator Value	Total Image Score	Image Difference
2.1.02 (San Diego)	32,162.8154296875	71,721.32638487135	103,884.14181455885	30,541.015625	105.31023533925945
3.2.25 (Pentagon)	60,915.33544921875	60,873.01708984375	121,788.3525390625	67,593.99462890625	90.1194489002261
wash-ir (Washington, D.C.)	141,301.09555555554	322,853.4367403431	464,154.53229589865	142,055.51133333333	99.46892889216558

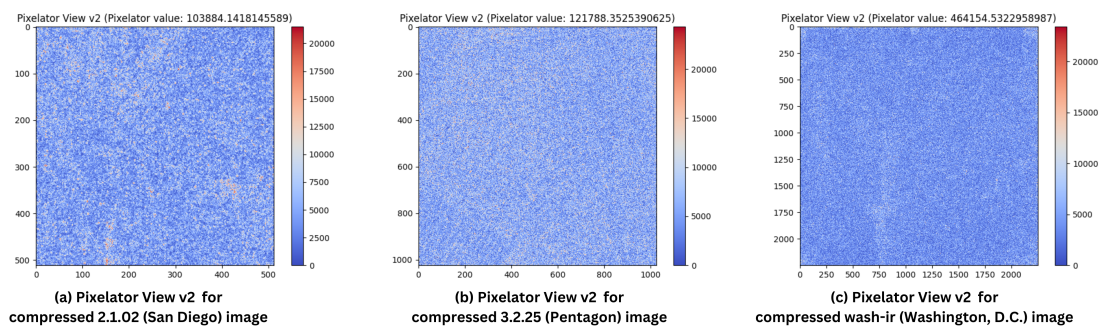


Figure 7. Pixelator View v2 for respective compressed images using JPEG compression algorithm at 85% quality: (a) 2.1.02 (Colour Image of San Diego, 512 × 512); (b) 3.2.25 (Grey Image of Pentagon, 1024 × 1024); and (c) wash-ir (Infra-red image of Washington, D.C.) from “Volume 2: Aerials” of the USC-SIPI Image Database [49].

4.6. Experiment Set 5: Comparative Study

In this comparative study, Pixelator v2’s performance was compared with that of Pixelator, MSE, SSIM, Q, FSIM, LPIPS, FID, and DISTs (as already discussed in Section 2). One random pixel in the R channel was altered by an increment of 1 (as shown in Section 4.2), and Table 6 summarises the results of the comparative study, demonstrating Pixelator v2’s superior ability to detect such changes. Figure 8 shows the differences in output from SSIM Map and Pixelator View v2 respectively. Additionally, from Table 6 it should be noted that Pixelator v2 value and Pixelator value for respective images are the same as in Pixelator View v2, Pixelator LAB Value in Pixelator Value (as shown in Equation (3)) is 0.0 (as confirmed in Table 2), hence, only the computed value of Pixelator RGB Value is shown in the output (Pixelator Value).

Table 6. Comparative study of Pixelator v2 with other methods - Pixelator [21], MSE [18], SSIM [20], Q [19], FSIM [35], LPIPS [30], FID [31] and DISTs [32].

Image	Pixelator v2	Pixelator [21]	MSE [18]	SSIM [20]	Q [19]	FSIM [35]	LPIPS [30]	FID [31]	DISTs [32]
2.1.02 (San Diego)	0.0009765625	0.0009765625	0.0	1.0	1.0	1.0	$1.5085080407040152 \times 10^{-10}$	$2.7785652667589034 \times 10^{-07}$	$2.434088169359554 \times 10^{-07}$
3.2.25 (Pentagon)	0.00048828125	0.00048828125	0.0	1.0	1.0	1.0	$5.120944224223933 \times 10^{-10}$	$2.7785652667589034 \times 10^{-07}$	$2.434088169359554 \times 10^{-07}$
wash-ir (Washington, D.C.)	0.0002222222222222223	0.0002222222222222223	0.0	1.0	0.9999999999999999	1.0	$3.51561280087509 \times 10^{-11}$	$2.7785652667589034 \times 10^{-07}$	$2.434088169359554 \times 10^{-07}$

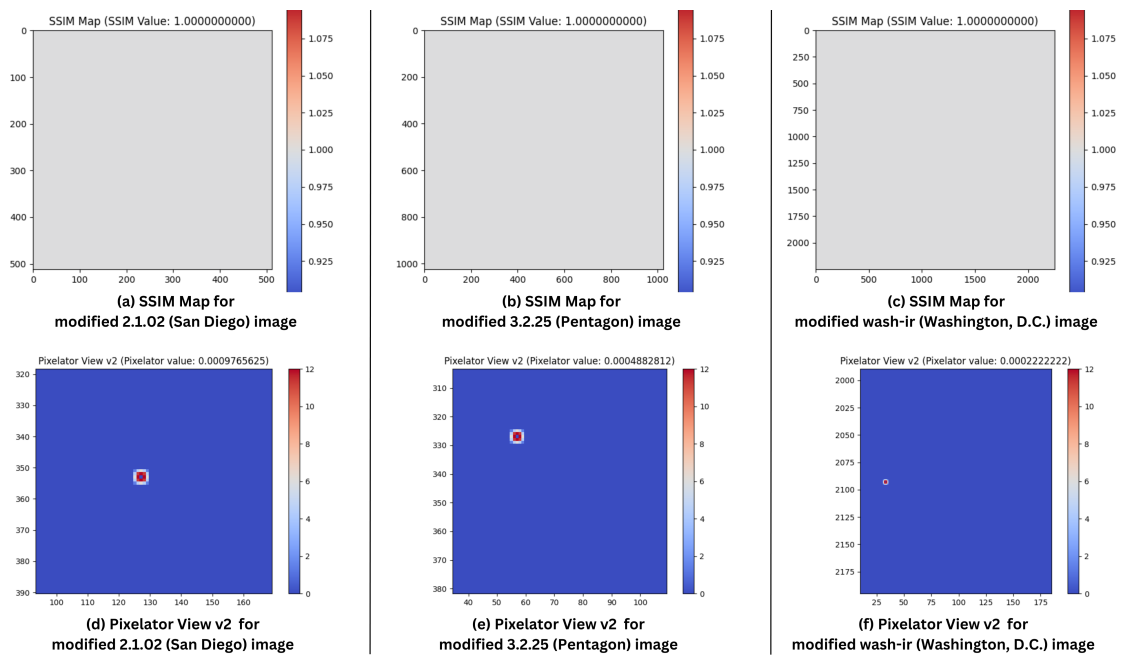


Figure 8. SSIM Map and Pixelator View v2 for respective modified images for one-pixel alteration in R channel: (a) SSIM Map of 2.1.02 (Colour Image of San Diego, 512×512); (b) SSIM Map of 3.2.25 (Grey Image of Pentagon, 1024×1024); (c) SSIM Map of wash-ir (Infra-red image of Washington, D.C.); (d) Pixelator View v2 of 2.1.02 (Colour Image of San Diego, 512×512); (e) Pixelator View v2 of 3.2.25 (Grey Image of Pentagon, 1024×1024); (f) Pixelator View v2 of wash-ir (Infra-red image of Washington, D.C.). *Note:* In (c), the SSIM value is rounded off from 0.9999999999999999 to 1.0 as part of computing the SSIM Map display.

It can be noticed that the consistent results produced by MSE, SSIM, Q, FID, and DISTs across multiple image comparisons in Table 6 can be attributed to their inherent design limitations in capturing subtle variations. MSE computes pixel-wise differences without considering perceptual factors, which results in uniform outputs when pixel-level differences are minimal. Similarly, SSIM and Q, while accounting for structural information and contrast, still rely heavily on pixel-level operations, failing to reflect finer perceptual differences that exist between images. FID and DISTs, which operate on high-level feature representations derived from neural networks, also show limited sensitivity to minor changes in pixel intensities. This results in uniform scores, as these metrics focus more on global features and textures rather than pixel-level nuances. On the other hand, LPIPS leverages deep network features that are more perceptually aligned with human vision, enabling it to capture subtle variations across different images. Unlike FID and DISTs, LPIPS is sensitive to even minor perturbations in local regions, explaining the varying values for each image comparison in Table 6.

In contrast, Pixelator v2, by combining pixel-wise analysis in the RGB space with perceptual insights from the CIE-LAB colour space, is able to capture both low-level pixel alterations and perceptual differences, thereby outperforming traditional and neural network-based metrics. This comprehensive approach ensures that even subtle variations, such as one-pixel changes, are accurately detected and reflected in the results.

In summary, Pixelator v2 demonstrates superiority over these methods due to its ability to combine pixel-level analysis in the RGB space with perceptual analysis in the CIE-LAB colour space. By employing Sobel edge detection, Pixelator v2 enhances structural comparison, detecting even minute pixel modifications more effectively than the other approaches. This multi-faceted approach, coupled with visualisation (Pixelator View v2), makes Pixelator v2 more robust in tamper detection, watermark analysis, and subtle image changes.

4.7. Experiment Set 6: Additional Evaluation Using KADID-10k Dataset

In addition to using images from the USC-SIPI Image Database, we further evaluate Pixelator v2 using the KADID-10k dataset [50]. This dataset includes 81 reference images and 10,125 distorted images (resulting from 25 types of distortion applied at five levels each). The diversity of distortions in KADID-10k allows for comprehensive testing of Pixelator v2's robustness against a wide range of visual degradations. For this experiment, we have selected the first level (Distortion Level 01) of each distortion type as it represents the lowest distortion level, providing an effective benchmark for evaluating Pixelator v2's sensitivity to subtle alterations. The selected distortions for additional experimentation are:

- **Lens Blur** (Distortion Type 02): Common in digital image forgeries, lens blur can obscure details that may hide tampering. Testing Pixelator v2 on this distortion type highlights its ability to detect subtle changes even under realistic, non-uniform blurring conditions.
- **Colour Shift** (Distortion Type 05): Colour shifts are often used in tampered images to alter their aesthetic or convey false information. This distortion tests Pixelator v2's sensitivity to chromatic changes crucial for identifying visual inconsistencies in tampered images.
- **JPEG Compression** (Distortion Type 10): JPEG compression, frequently used in digital communications, introduces lossy artefacts. Pixelator v2's detection of compression artefacts ensures image integrity checks, essential for secure media transmission.
- **White Noise in Colour Component** (Distortion Type 12): Often encountered in low-quality digital images, Gaussian noise can obscure fine details. Pixelator v2's performance with this distortion tests its robustness in identifying tampering despite random noise.
- **High Sharpen** (Distortion Type 24): Enhancing sharpness emphasizes edges, which is a technique sometimes used in forged images to create emphasis on specific parts. Pixelator v2's capability to detect structural modifications due to sharpening tests its effectiveness in highlighting suspicious edits.

Each selected distortion category represents distinct types of alterations—blur, colour shifts, compression artefacts, noise, and sharpening—that Pixelator v2 must accurately identify to demonstrate its effectiveness across various image quality challenges. By incorporating these distortions, our experiments thoroughly assess Pixelator v2's sensitivity to diverse visual changes relevant to image security, tamper detection, and quality assessment contexts. The results of this evaluation are summarized in Table 7, demonstrating the efficacy of Pixelator v2.

The KADID-10k dataset also provides subjective scores, allowing us to measure the correlation between Pixelator v2's output and human-perceived quality. We calculate Spearman's correlation coefficient [61] to quantify Pixelator v2's alignment with subjective scores across different distortion types. To assess the alignment of Pixelator v2 with human subjective perception, we computed Spearman's correlation coefficient, a widely used metric for evaluating perceptual similarity alignment in image quality assessments. Spearman's correlation was chosen due to its robustness in evaluating rank-based consistency, making it suitable for comparing subjective scores, which inherently represent perceptual ranks, with our metric scores. This method quantifies how well the ranking of images by Pixelator v2 corresponds to human judgment, capturing the perceptual sensitivity of the proposed method. Spearman's correlation values for Pixelator v2 and other metrics on the KADID-10k dataset are presented in Table 8. Table 8 presents these correlations, along with a comparison to other metrics of widely used state-of-the-art methods such as MSE, SSIM, FSIM, LPIPS, FID and DeepFL-IQA [62]. DeepFL-IQA is intrinsically linked to the KADID-10k dataset, as it utilizes a weakly-supervised approach to learn quality assessment features specifically tuned to image distortions. This method leverages the subjective scores within KADID-10k to train a feature-extraction network that aligns well with human perception, aiming for a high correlation with subjective quality ratings. By incorporating

DeepFL-IQA in our comparisons, we benchmark Pixelator v2 against a metric designed to reflect perceptual relevance closely, enabling us to evaluate how well our approach aligns with established human-centric quality assessments.

Table 7. Results of Pixelator v2 on selected distortions from the KADID-10k dataset. Each reference and distorted image is labeled accordingly.

Reference Image	Distorted Image	Pixelator Value	Image Difference
		69,418.3653287521	88.0778355972
		70,751.1185728180	84.4637466036
		83,082.3003352856	113.5062603274
		85,321.0182552117	90.2684083904
		54,537.2469911299	99.2288520733

Image Captions: (1) Reference Image: I09.png, Distorted Image: Lens blurred image at level 1—I09_02_01.png; (2) Reference Image: I02.png, Distorted Image: Colour shifted image at level 1—I02_05_01.png; (3) Reference Image: I03.png, Distorted Image: JPEG compressed image at level 1—03_10_01.png; (4) Reference Image: I04.png, Distorted Image: White noise in colour component image at level 1—I04_12_01.png; (5) Reference Image: I06.png, Distorted Image: High sharpened image at level 1—I06_24_01.png.

The selected distortions include Lens Blur, Colour Shift, JPEG Compression, White Noise, and High Sharpen (as mentioned above), chosen to reflect common alterations in security contexts. The following table summarises Pixelator v2’s performance, including correlation values, perceptual sensitivity, and computational efficiency. To calculate the Spearman’s correlation values presented in Table 8, we evaluated the output scores of each metric across the KADID-10k dataset and computed their correlation with the subjective scores provided within the dataset. Spearman’s correlation assesses the rank-based consistency between the metric’s scores and human-perceived quality, capturing perceptual

alignment. Each value in Table 8 (for example, 0.45 for MSE, 0.62 for SSIM, etc.) represents the correlation coefficient obtained from this comparison. This method enables us to quantify how well each metric aligns with subjective quality ratings, particularly for subtle distortions, making it suitable for evaluating perceptual similarity.

Table 8. Comparison of Pixelator v2 with State-of-the-Art Metrics on KADID-10k Dataset.

Method	Correlation (Spearman)	Perceptual Sensitivity	Structural Analysis	Computational Efficiency
MSE [18]	0.45	Low	None	High
SSIM [20]	0.62	Moderate	Limited	Moderate
FSIM [35]	0.78	High	Moderate	Low
LPIPS [30]	0.80	High	None	Low
FID [31]	0.75	High	None	Low
DeepFL-IQA [62]	0.82	Very High	High	Low
Pixelator v2 (proposed)	0.69	High	High	High

The correlation values presented in Table 8 reflect the degree to which each metric aligns with human-perceived image quality, as measured by subjective scores in the KADID-10k dataset.

For example, traditional methods such as MSE and SSIM achieve correlation scores of 0.45 and 0.62, respectively, reflecting moderate alignment with subjective scores. While these metrics are effective in detecting intensity-based differences, they lack perceptual sensitivity, limiting their correlation with subjective human assessments. In contrast, metrics such as FSIM (0.78) and LPIPS (0.80), which integrate feature-based and deep learning approaches, respectively, demonstrate higher correlations due to their perceptual sensitivity, which better aligns with human visual judgment.

Our proposed Pixelator v2 achieves a correlation score of 0.69, indicating robust alignment with subjective assessments, particularly when considering its computational efficiency. Although not as high as deep learning-based metrics such as DeepFL-IQA (0.82), Pixelator v2 maintains a high degree of perceptual and structural sensitivity while offering the practical advantage of high computational efficiency, making it well-suited for real-time and resource-constrained applications. This balance of accuracy and efficiency highlights Pixelator v2's suitability for applications in image security and tamper detection.

Note: To ensure the robustness of the correlation values presented in Table 8, we computed Spearman's correlation coefficient across the entire KADID-10k dataset and subsets of selected distortions, aligning Pixelator v2's metric scores with subjective quality scores. The calculated Spearman's correlation values for each metric, such as 0.45 for MSE, 0.62 for SSIM, and 0.69 for Pixelator v2, represent the rank-order consistency between the metric's scores and the subjective quality scores. This approach enables an objective evaluation of how well Pixelator v2 and other metrics reflect perceptual alignment in image quality. Additionally, it should be kept in mind that while Pearson's correlation [63] assesses linear relationships, Spearman's correlation is more suited for evaluating rank-based consistency, as it captures non-linear associations between subjective quality scores and metric outputs. Given the subjective nature of image quality assessments, which represent perceptual ranks rather than absolute linear relationships, Spearman's correlation was chosen to provide a robust measure of how well Pixelator v2 aligns with human perception.

In summary, the results across all experiments (Sections 4.2–4.7) demonstrate the effectiveness of Pixelator v2 in detecting subtle image modifications, outperforming conventional and state-of-the-art methods in various scenarios.

5. Discussion and Future Works

The Pixelator value in Pixelator v2, while expressed as a single scalar metric, does not operate within the same normalized scale as SSIM or MSE. Unlike SSIM index, which is a decimal value between -1 and 1 —where 1 indicates perfect similarity, 0 indicates no similarity, and -1 indicates perfect anti-correlation; Pixelator v2 provides an unbounded score that reflects the aggregated pixel-wise differences in both the RGB and CIE-LAB colour spaces. Pixelator value in Pixelator v2 alone does not hold inherent significance unless interpreted in context. This means that the score reflects the cumulative differences across the image's dimensions.

Moreover, as the Pixelator v2 tool evolves, future iterations could explore more intuitive thresholds or normalizations that provide a clearer indication of the extent of change, akin to SSIM's $[-1, 1]$ scale. However, it is essential to note that Pixelator v2's strength lies in its ability to capture both minute pixel-level alterations and perceptual differences across channels, both quantitatively and visually, making it inherently more sensitive than traditional metrics. This sensitivity, though not directly comparable to SSIM's simplicity, offers greater flexibility in identifying image manipulations across a wide range of applications.

While SSIM and MSE, when used in combination, attempt to address each other's limitations—SSIM focusing on structural and perceptual information, and MSE concentrating on pixel-wise intensity differences—there are inherent limitations in how these methods operate individually and together. SSIM's reliance on luminance, contrast, and structure, and MSE's pixel-based analysis, still overlook certain subtle pixel-level and perceptual alterations, particularly in non-perceptually significant regions. The combination of these methods may provide a more balanced measure of image similarity, but their individual strengths and weaknesses are constrained to specific domains.

Pixelator v2, on the other hand, outperforms this combined approach by integrating pixel-level analysis in the RGB colour space with perceptual evaluation in the CIE-LAB colour space, allowing it to capture both the low-level pixel differences that MSE detects and the perceptual insights that SSIM seeks to provide. Furthermore, the inclusion of Sobel edge detection enhances its capability to identify even minor structural changes. This comprehensive and multi-faceted approach enables Pixelator v2 to be more sensitive to image alterations, offering superior detection accuracy, particularly in high-stakes applications such as tamper detection and image security. Therefore, Pixelator v2 provides a more holistic and robust framework for image comparison than the combination of SSIM and MSE.

Pixelator v2 represents a significant advancement over previous methodologies in image comparison, particularly in security-sensitive contexts such as tamper detection and steganography. Moreover, Pixelator v2 provides a nuanced and robust approach to detecting subtle changes that traditional methods, such as MSE, SSIM, and Q, often overlook. However, despite its strengths, Pixelator v2 does have certain limitations that present opportunities for future research and enhancement.

One notable shortcoming of Pixelator v2 lies in its reliance on predefined metrics, which may not generalise optimally across all types of images or application domains. For example, while the combination of RGB and LAB spaces provides strong perceptual and structural analysis, the method does not adapt dynamically to varying image types or contexts where specific features might be more significant. This limitation suggests the need for future versions of Pixelator to incorporate machine learning models that can be trained on specific image domains. By leveraging domain-specific knowledge, such models could assist in evaluating the significance of differences, prioritising those that are more meaningful in certain contexts. For instance, in medical imagery, small anomalies can be critical, and a machine learning-based extension of Pixelator v2 could help identify these changes more accurately by weighting features of interest according to their relevance within the medical domain.

Moreover, Pixelator v2's current design is best suited for static image comparison, which limits its application in scenarios where temporal changes are significant, such as

video surveillance or time-lapse analysis. Expanding Pixelator v2 to support temporal analysis would enable it to be used in more complex scenarios, such as video security, where the evolution of differences across frames may provide important insights. Future versions could integrate a time-based view that highlights how discrepancies evolve across image sequences, making the tool more suitable for video surveillance, anomaly detection, and other security-sensitive applications involving image sequences.

Another area for potential improvement is the computational efficiency of the method. Pixelator v2's computational efficiency is achieved by combining LAB colour space analysis and Sobel edge detection, allowing perceptual and structural evaluation without relying on deep feature extraction or extensive computational resources. Unlike FSIM, LPIPS, and PieAPP, which involve feature extraction through complex transformations or neural networks, Pixelator v2's streamlined approach emphasizes real-time applicability in security contexts, making it faster and more adaptable for practical use. This distinction underscores Pixelator v2's unique advantage in balancing perceptual accuracy with computational speed. While Pixelator v2 is relatively efficient compared to deep learning-based approaches, the inclusion of additional features, such as machine learning models or temporal analysis, will require further optimisation to maintain real-time performance. Techniques such as model pruning, quantisation, or using more efficient algorithms for edge detection and perceptual analysis could be explored to ensure that the enhanced capabilities do not come at the cost of computational feasibility.

In summary, while Pixelator v2 offers a powerful solution for image comparison, there are multiple avenues for future work. Integrating machine learning for contextual evaluation and extending the tool for temporal analysis are promising directions. These extensions would not only broaden the applicability of Pixelator v2 across different domains but also improve its adaptability and accuracy in detecting meaningful differences in complex image sets.

6. Conclusions

In this paper, we introduced Pixelator v2, a novel perceptual image comparison method that integrates pixel-wise RGB analysis, perceptual relevance through the CIE-LAB colour space, and structural integrity via Sobel edge detection. By addressing the limitations of traditional methods such as MSE, SSIM, and even advanced techniques such as LPIPS, FID and DISTs, Pixelator v2 offers a robust, computationally efficient, and perceptually aligned approach to image comparison. Experimental results demonstrate that Pixelator v2 excels in detecting both subtle pixel-level changes and perceptually significant modifications, outperforming state-of-the-art techniques in a variety of test scenarios. Pixelator v2's dual approach—merging pixel-level precision with perceptual and structural insights—positions it as a powerful tool for image security, tamper detection, and forensic analysis.

7. Code Availability

The source-code for Pixelator v2 is available from <https://github.com/somdipdey/Pixelator-View-v2> (accessed on 18 November 2024) and the source-code for Pixelator [21] is available from <https://github.com/somdipdey/Pixelator-View> (accessed on 18 November 2024).

Author Contributions: Conceptualization, S.D. and S.S.; methodology, S.D.; software, S.D.; validation, S.D., J.A.A.-A., A.B., S.S., R.P., S.H. and J.T.; formal analysis, S.D.; investigation, S.D.; resources, S.D.; data curation, S.D.; writing—original draft preparation, S.D.; writing—review and editing, S.D., J.A.A.-A., A.B., S.S., R.P., S.H. and J.T.; visualization, S.D.; supervision, S.D.; project administration, S.D.; funding acquisition, S.D. and S.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by Nosh Technologies under Grant nosh/computing-tech-000001.

Data Availability Statement: The source-code and data for Pixelator v2 is available from <https://github.com/somdipdey/Pixelator-View-v2> (accessed on 18 November 2024).

Conflicts of Interest: Authors Suman Saha and Rohit Purkait were employed by the company Nosh Technologies. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Korus, P. Digital image integrity—A survey of protection and verification techniques. *Digit. Signal Process.* **2017**, *71*, 1–26. [CrossRef]
2. Du, L.; Ho, A.T.; Cong, R. Perceptual hashing for image authentication: A survey. *Signal Process. Image Commun.* **2020**, *81*, 115713. [CrossRef]
3. Dey, S. SD-EI: A cryptographic technique to encrypt images. In Proceedings of the 2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensic (CyberSec), Kuala Lumpur, Malaysia, 26–28 June 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 28–32.
4. Mohammad, O.F.; Rahim, M.S.M.; Zeebaree, S.R.M.; Ahmed, F. A survey and analysis of the image encryption methods. *Int. J. Appl. Eng. Res.* **2017**, *12*, 13265–13280.
5. Dey, S.; Mondal, K.; Nath, J.; Nath, A. Advanced steganography algorithm using randomized intermediate QR host embedded with any encrypted secret message: ASA_QR algorithm. *Int. J. Mod. Educ. Comput. Sci.* **2012**, *4*, 59. [CrossRef]
6. Chennamma, H.; Madhushree, B. A comprehensive survey on image authentication for tamper detection with localization. *Multimed. Tools Appl.* **2023**, *82*, 1873–1904. [CrossRef]
7. Dey, S.; Nath, J.; Nath, A. An Integrated Symmetric Key Cryptographic Method-Amalgamation of TTJSA Algorithm, Advanced Caesar Cipher Algorithm, Bit Rotation and Reversal Method: SJA Algorithm. *Int. J. Mod. Educ. Comput. Sci.* **2012**, *4*, 1. [CrossRef]
8. Birajdar, G.K.; Mankar, V.H. Digital image forgery detection using passive techniques: A survey. *Digit. Investig.* **2013**, *10*, 226–245. [CrossRef]
9. Dey, S. SD-AEI: An advanced encryption technique for images. In Proceedings of the 2012 Second International Conference on Digital Information Processing and Communications (ICDIPC), Klaipeda, Lithuania, 10–12 July 2012.
10. Dey, S. An image encryption method: SD-advanced image encryption standard: SD-AIES. *Int. J.-Cyber-Secur. Digit. Forensics* **2012**, *1*, 82–88.
11. Johnson, N.F.; Duric, Z.; Jajodia, S. *Information Hiding: Steganography and Watermarking—Attacks and Countermeasures: Steganography and Watermarking: Attacks and Countermeasures*; Springer Science & Business Media: Berlin, Germany, 2001; Volume 1.
12. Fridrich, J. *Steganography in Digital Media: Principles, Algorithms, and Applications*; Cambridge University Press: Cambridge, UK, 2009.
13. Hussain, M.; Hussain, M. A survey of image steganography techniques. *Int. J. Adv. Sci. Technol.* **2013**, *54*, 113–124.
14. Chang, C.C.; Lin, C.Y. Reversible steganography for VQ-compressed images using side matching and relocation. *IEEE Trans. Inf. Forensics Secur.* **2006**, *1*, 493–501. [CrossRef]
15. Barni, M.; Bartolini, F.; Piva, A. Improved wavelet-based watermarking through pixel-wise masking. *IEEE Trans. Image Process.* **2001**, *10*, 783–791. [CrossRef] [PubMed]
16. Mehta, R.; Rajpal, N.; Vishwakarma, V.P. A robust and efficient image watermarking scheme based on Lagrangian SVR and lifting wavelet transform. *Int. J. Mach. Learn. Cybern.* **2017**, *8*, 379–395. [CrossRef]
17. Langelaar, G.C.; Setyawan, I.; Lagendijk, R.L. Watermarking digital image and video data. A state-of-the-art overview. *IEEE Signal Process. Mag.* **2000**, *17*, 20–46. [CrossRef]
18. Martens, J.B.; Meesters, L. Image dissimilarity. *Signal Process.* **1998**, *70*, 155–176. [CrossRef]
19. Wang, Z.; Bovik, A.C. A universal image quality index. *IEEE Signal Process. Lett.* **2002**, *9*, 81–84. [CrossRef]
20. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef]
21. Dey, S.; Singh, A.K.; Prasad, D.K.; McDonald-Maier, K.D. Socodecnn: Program source code for visual cnn classification using computer vision methodology. *IEEE Access* **2019**, *7*, 157158–157172. [CrossRef]
22. Dey, S. Novel DVFS Methodologies for Power-Efficient Mobile MPSoC. Ph.D. Thesis, University of Essex, Colchester, UK, 2023.
23. Fairchild, M.D. *Color Appearance Models*; John Wiley & Sons: Hoboken, NJ, USA, 2013.
24. Schanda, J. *Colorimetry: Understanding the CIE System*; John Wiley & Sons Press: Hoboken, NJ, USA, 2007.
25. Sobel, I.; Feldman, G. A 3×3 isotropic gradient operator for image processing. *Talk Stanf. Artif. Proj.* **1968**, *1968*, 271–272.
26. Kanopoulos, N.; Vasanthavada, N.; Baker, R.L. Design of an image edge detection filter using the Sobel operator. *IEEE J. Solid-State Circuits* **1988**, *23*, 358–367. [CrossRef]
27. Aybar, E. *Sobel Edge Detection Method for Matlab*; Anadolu University, Porsuk Vocational School: Eskişehir, Turkey, 2006; Volume 26410.
28. Gao, W.; Zhang, X.; Yang, L.; Liu, H. An improved Sobel edge detection. In Proceedings of the 2010 3rd International Conference on Computer Science and Information Technology, Chengdu, China, 9–11 July 2010; IEEE: Piscataway, NJ, USA, 2010; Volume 5, pp. 67–71.

29. Canny, J. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1986**, PAMI-8, 679–698. [[CrossRef](#)]
30. Zhang, R.; Isola, P.; Efros, A.A.; Shechtman, E.; Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 586–595.
31. Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; Hochreiter, S. GANs trained by a two time-scale update rule converge to a local nash equilibrium. *Adv. Neural Inf. Process. Syst.* **2017**, 30, 6629–6640.
32. Ding, K.; Ma, K.; Wang, S.; Simoncelli, E.P. Image quality assessment: Unifying structure and texture similarity. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, 44, 2567–2581. [[CrossRef](#)] [[PubMed](#)]
33. Mayyas, M. Image Reconstruction and Evaluation: Applications on Micro-Surfaces and Lenna Image Representation. *J. Imaging* **2016**, 2, 27. [[CrossRef](#)]
34. Aljanabi, M.A.; Hussain, Z.M.; Shnain, N.A.A.; Lu, S.F. Design of a hybrid measure for image similarity: A statistical, algebraic, and information-theoretic approach. *Eur. J. Remote Sens.* **2019**, 52, 2–15. [[CrossRef](#)]
35. Zhang, L.; Zhang, L.; Mou, X.; Zhang, D. FSIM: A feature similarity index for image quality assessment. *IEEE Trans. Image Process.* **2011**, 20, 2378–2386. [[CrossRef](#)]
36. Wang, K.; Gou, C.; Duan, Y.; Lin, Y.; Zheng, X.; Wang, F.Y. Generative adversarial networks: Introduction and outlook. *IEEE/CAA J. Autom. Sin.* **2017**, 4, 588–598. [[CrossRef](#)]
37. Saxena, D.; Cao, J. Generative adversarial networks (GANs) challenges, solutions, and future directions. *ACM Comput. Surv. (CSUR)* **2021**, 54, 1–42. [[CrossRef](#)]
38. Gui, J.; Sun, Z.; Wen, Y.; Tao, D.; Ye, J. A review on generative adversarial networks: Algorithms, theory, and applications. *IEEE Trans. Knowl. Data Eng.* **2021**, 35, 3313–3332. [[CrossRef](#)]
39. Maji, P.; Mullins, R. On the reduction of computational complexity of deep convolutional neural networks. *Entropy* **2018**, 20, 305. [[CrossRef](#)]
40. He, K.; Sun, J. Convolutional neural networks at constrained time cost. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5353–5360.
41. Dey, S.; Singh, A.K.; Prasad, D.K.; McDonald-Maier, K.D. Iron-man: An approach to perform temporal motionless analysis of video using crn in mp soc. *IEEE Access* **2020**, 8, 137101–137115. [[CrossRef](#)]
42. Dey, S.; Singh, A.K.; Prasad, D.K.; McDonald-Maier, K. Temporal motionless analysis of video using crn in mp soc. In Proceedings of the 2020 IEEE 31st International Conference on Application-Specific Systems, Architectures and Processors (ASAP), Manchester, UK, 6–8 July 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 73–76.
43. Neshatpour, K.; Behnia, F.; Homayoun, H.; Sasan, A. ICNN: An iterative implementation of convolutional neural networks to enable energy and computational complexity aware dynamic approximation. In Proceedings of the 2018 Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden, Germany, 19–23 March 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 551–556.
44. Kong, C.; Luo, A.; Wang, S.; Li, H.; Rocha, A.; Kot, A.C. Pixel-inconsistency modeling for image manipulation localization. *arXiv* **2023**, arXiv:2310.00234.
45. Prashnani, E.; Cai, H.; Mostofi, Y.; Sen, P. Pieapp: Perceptual image-error assessment through pairwise preference. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1808–1817.
46. Deza, E.; Deza, M.M.; Deza, M.M.; Deza, E. *Encyclopedia of Distances*; Springer: Berlin/Heidelberg, Germany, 2009.
47. Malkauthekar, M. Analysis of Euclidean distance and Manhattan distance measure in face recognition. In Proceedings of the Third International Conference on Computational Intelligence and Information Technology (CIIT 2013), Mumbai, India, 18–19 October 2013; IET: Stevenage, UK, 2013; pp. 503–507.
48. Bishop, C.M.; Nasrabadi, N.M. *Pattern Recognition and Machine Learning*; Springer: Berlin/Heidelberg, Germany, 2006; Volume 4.
49. Weber, A.G. The USC-SIPI Image Database: Version 5. 2006. Available online: <http://sipi.usc.edu/database/> (accessed on 18 November 2024).
50. Lin, H.; Hosu, V.; Saupe, D. KADID-10k: A Large-scale Artificially Distorted IQA Database. In Proceedings of the 2019 Tenth International Conference on Quality of Multimedia Experience (QoMEX), Berlin, Germany, 5–7 June 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–3.
51. Bansal, K.; Agrawal, A.; Bansal, N. A survey on steganography using least significant bit (lsb) embedding approach. In Proceedings of the 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), Tirunelveli, India, 15–17 June 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 64–69.
52. Rachael, O.; Misra, S.; Ahuja, R.; Adewumi, A.; Ayeni, F.; Mmaskeliunas, R. Image steganography and steganalysis based on least significant bit (LSB). In Proceedings of the ICETIT 2019: Emerging Trends in Information Technology, Delhi, India, 15–16 December 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 1100–1111.
53. Anwar, F.; Fadlil, A.; Riadi, I. Image Quality Analysis of PNG Images on WhatsApp Messenger Sending. *Telematika* **2021**, 14, 1–12. [[CrossRef](#)]
54. Mayukha, S.; Sundaresan, M. Enhanced Image Compression Technique to Improve Image Quality for Mobile Applications. In *Rising Threats in Expert Applications and Solutions: Proceedings of FICR-TEAS 2020*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 281–291.

55. Douglas, Z. Digital Image Recompression Analysis of Instagram. Master's Thesis, University of Colorado at Denver, Denver, CO, USA, 2018.
56. Wang, Z.; Guo, H.; Zhang, Z.; Song, M.; Zheng, S.; Wang, Q.; Niu, B. Towards compression-resistant privacy-preserving photo sharing on social networks. In Proceedings of the Twenty-First International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing, Virtual Event, 11–14 October 2020; pp. 81–90.
57. Sharma, M.; Peng, Y. How visual aesthetics and calorie density predict food image popularity on instagram: A computer vision analysis. *Health Commun.* **2024**, *39*, 577–591. [[CrossRef](#)] [[PubMed](#)]
58. Triantaphillidou, S.; Allen, E. Digital image file formats. In *The Manual of Photography*; Routledge: London, UK, 2012; pp. 315–328.
59. Dheemanth, H. LZW data compression. *Am. J. Eng. Res.* **2014**, *3*, 22–26.
60. Badshah, G.; Liew, S.C.; Zain, J.M.; Ali, M. Watermark compression in medical image watermarking using Lempel-Ziv-Welch (LZW) lossless compression technique. *J. Digit. Imaging* **2016**, *29*, 216–225. [[CrossRef](#)]
61. Spearman, C. The Proof and Measurement of Association Between Two Things. 1961. Available online: <https://psycnet.apa.org/record/2006-10257-005> (accessed on 15 November 2024).
62. Lin, H.; Hosu, V.; Saupe, D. DeepFL-IQA: Weak Supervision for Deep IQA Feature Learning. *arXiv* **2020**, arXiv:2001.08113.
63. Pearson, K. VII. Note on regression and inheritance in the case of two parents. *Proc. R. Soc. Lond.* **1895**, *58*, 240–242.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.