

Article

Not peer-reviewed version

LLMSeamCarver: LLM-Enhanced Content-Aware Image Resizing

[Dong Liu](#)*, Yanxuan Yu, [Lianghao Tan](#), Wenjun Wu, Bide Zhao, Bingjie Lu

Posted Date: 6 January 2025

doi: 10.20944/preprints202412.1897.v4

Keywords: efficient machine learning; LLMs in vision



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

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.

Article

LLMSeamCarver: LLM-Enhanced Content-Aware Image Resizing

Dong Liu ¹, Yanxuan Yu ², Lianghao Tan ³, Wenjun Wu ⁴, Bide Zhao ⁵ and Bingjie Lu ⁶

¹ Yale University

² Columbia University

³ Arizona State University

⁴ UIUC

⁵ University of Sydney

⁶ Rutgers University

Abstract: This paper introduces "LLMSeamCarver," a LLM-enhanced methodology for image resizing. LLMSeamCarver addresses the limitations of traditional seam carving with static pre-defined parameters, it uses LLM to achieve dynamic and user-controlled dynamically-resizing of images. The inclusion of LLMs in this research facilitates dynamic optimization of parameter tuning and adaptive energy function adjustments, enhancing overall robustness and efficiency of image resizing. LLMSeamCarver emerges as a transformative tool, offering versatile, high-quality resized images.

I. Introduction

LLMSeamCarver is an image resizing method designed to maintain important image details during resizing. By integrating LLMs, LLMSeamCarver goes beyond traditional methods, enabling a smarter, context-aware resizing process. LLMs enhance tasks such as region prioritization, interpolation, edge detection, and energy calculation by analyzing image context or textual inputs. This allows LLMSeamCarver to preserve crucial image features like faces and text while optimizing resizing for different scenarios.

This paper explores how LLMs improve the accuracy and efficiency of image resizing.

II. Background

i. Image Resizing and LLMs

LLMSeamCarver enhances the traditional seam carving method by incorporating LLMs. The integration of LLMs allows for dynamic resizing of images based on real-time context and user input, rather than relying on fixed parameters. The LLM dynamically optimizes the energy functions that determine how seams are selected and removed, making the resizing process adaptive and context-sensitive. This approach allows for superior preservation of image details and better quality when resizing for specific tasks, such as creating thumbnails or preparing images for various screen sizes.

LLMs contribute by adjusting the parameters based on the content of the image and the desired effect, improving the quality of resized images while providing flexibility. The dynamic optimization of the resizing process using LLMs represents a major leap in the flexibility and efficiency of image resizing techniques.

ii. LLM-Related Work

Recent research in LLMs has demonstrated their potential in a variety of domains, including text, image, and video generation.

Recently, LLMs has shown its power and potential to enhance traditional image processing workflows through advanced model architectures and optimization techniques. For instance, researchers

have developed methods that optimize convolutional neural network (CNN) layers for feature extraction in image processing tasks, thereby improving the performance of deep learning models in image classification and segmentation [2,22]. Zhang et al. (2024) explored the use of LLMs in multi-modal fusion networks, enabling the integration of both visual and textual information, which enhances image analysis tasks [10,11,24,26].

Additionally, efficient algorithm design [1,6,24,27,30] and efficient LLMs [17,19] have shown promising prospective in efficient model design with LLMs. Through dynamic optimization, LLMs allow for more context-aware resizing by adjusting energy functions during the process. This flexibility ensures that fine-grained details are preserved, which is especially crucial for tasks like content-aware resizing. Studies on text-to-image models have demonstrated how LLMs can modify images based on contextual prompts [7,13,15,16,28], providing further advancements in content-aware image resizing.

iii. Image and Vision Generation Work

The application of deep learning techniques in *image and vision generation* has seen remarkable progress in recent years [3,4,9,21,23,25,26]. Deep convolutional networks have been effectively utilized for texture classification, achieving high accuracy in distinguishing fine-grained patterns and structures (Bu et al., 2019) [1]. These approaches significantly enhance detail preservation during image resizing by maintaining textures and edges, ensuring that visual fidelity is retained across transformations.

Moreover, advances in *multi-modal fusion networks* and techniques for *image-driven predictions*—as demonstrated by Dan et al. (2024)—illustrate how artificial intelligence can process and modify images in real-time [3,14,26]. These methods integrate data from diverse sources, facilitating applications such as enhanced video editing and real-time object tracking.

Additionally, model compression is increasingly favored from both model optimization and system design perspectives, enabling efficient resource usage without sacrificing performance [8,18,20]. These innovations support dynamic, user-controlled visual generation, opening new possibilities for customizable and interactive media content creation.

iv. Image Resizing and Seam Carving Research

The traditional seam carving method was proposed by Avidan and Shamir (2007), other studies have contributed to enhancing seam carving methods. Kiess (2014) introduced improved edge preservation methods within seam carving [12], which is crucial for ensuring that resized images do not suffer from visible distortions along object boundaries. Zhang (2015) compared classic image resizing methods and found that seam carving provided superior results when compared to simpler resizing techniques, particularly in terms of detail preservation [29].

Frankovich (2011) further advanced seam carving by integrating energy gradient functionals to enhance the carving process, providing even more control over the resizing operation [5].

III. Functionality

LLMSeamCarver leverages LLM-Augmented methods to ensure adaptive, high-quality image resizing while preserving both structural and semantic integrity. The key functionalities are:

- **LLM-Augmented Region Prioritization:** LLMs analyze semantics or textual inputs to prioritize key regions, ensuring critical areas (e.g., faces, text) are preserved.
- **LLM-Augmented Bicubic Interpolation:** LLMs optimize bicubic interpolation for high-quality enlargements, adjusting parameters based on context or user input.
- **LLM-Augmented LC Algorithm:** LLMs adapt the LC algorithm by adjusting weights, ensuring the preservation of important image features during resizing.
- **LLM-Augmented Canny Edge Detection:** LLMs guide Canny edge detection to refine boundaries, enhancing clarity and accuracy based on contextual analysis.

- **LLM-Augmented Hough Transformation:** LLMs strengthen the Hough transformation, detecting structural lines and ensuring the preservation of geometric features.
- **LLM-Augmented Absolute Energy Function:** LLMs dynamically adjust energy maps to improve seam selection for more precise resizing.
- **LLM-Augmented Dual Energy Model:** LLMs refine energy functions, enhancing flexibility and ensuring effective seam carving across various use cases.

IV. LLM-Guided Region Prioritization

To enhance seam carving, we propose a formalized method where **LLMs** are used to compute semantic importance $S(x, y)$ for each pixel, directly modifying the energy map $E(x, y)$. This ensures content-aware resizing with minimal disruption to critical features.

i. Semantic Importance Scoring by LLMs

The semantic importance $S(x, y)$ is computed as:

$$S(x, y) = g(h(I), q(D)), \quad (1)$$

where:

- $h(I)$: Image embedding derived from input image I using a vision feature extractor.
- $q(D)$: Text embedding derived from optional user description D using a language transformer.
- g : A cross-modal scoring function combining $h(I)$ and $q(D)$, implemented via attention mechanisms.

If no description D is provided, $q(D)$ defaults to a generic embedding, allowing g to focus solely on image features.

ii. Energy Map Adjustment

The refined energy map $E'(x, y)$ integrates semantic importance into the standard energy formulation:

$$E'(x, y) = E(x, y) + \alpha \cdot S(x, y), \quad (2)$$

$$E(x, y) = \sqrt{(\nabla I_x(x, y))^2 + (\nabla I_y(x, y))^2},$$

where:

- $\nabla I_x(x, y), \nabla I_y(x, y)$: Gradients in x - and y -directions.
- $S(x, y)$: Semantic score indicating the importance of pixel (x, y) .
- α : Weighting factor balancing pixel-based energy and semantic importance.

The function $S(x, y)$ ensures that higher semantic scores reduce the likelihood of important regions being removed during seam carving.

iii. Semantic Score Calculation with LLMs

To compute $S(x, y)$, the LLM processes image embeddings $h(I)$ and textual embeddings $q(D)$:

$$h(I) = \text{VisionEncoder}(I), \quad (3)$$

$$q(D) = \text{TextEncoder}(D), \quad (4)$$

$$S(x, y) = \text{Attention}(h(I), q(D)), \quad (5)$$

where:

- **VisionEncoder:** Extracts regional features from I (e.g., object locations, edges).
- **TextEncoder:** Encodes user-provided descriptions into contextual embeddings.

- Attention: Combines $h(I)$ and $q(D)$ to assign $S(x, y)$ based on pixel relevance. The score $S(x, y)$ is normalized to a range $[0, 1]$ for compatibility with $E(x, y)$.

iv. Cumulative Energy Map Update

The cumulative energy map $E'(x, y)$ propagates semantic adjustments through the seam carving process. For a given pixel (i, j) , the update is:

$$E'(i, j) = E(i, j) + \alpha \cdot S(i, j) + \min \left\{ \begin{aligned} &C_L(i, j) + E'(i - 1, j - 1), \\ &C_R(i, j) + E'(i - 1, j + 1), \\ &C_U(i, j) + E'(i - 1, j) \end{aligned} \right\}, \quad (6)$$

where:

- C_L, C_R, C_U : Left, right, and upward cost terms adjusted with $S(x, y)$.
- min: Ensures the optimal seam path minimizes distortion of high-priority regions.

V. LLM-Augmented Bicubic Interpolation

The **LLM-Augmented Bicubic Interpolation** method enhances traditional bicubic interpolation by incorporating semantic importance scores $S(x, y)$ derived from **LLMs**. This approach ensures content-aware resizing, prioritizing regions of high semantic value such as faces, text, and objects.

i. Traditional Bicubic Interpolation

In standard bicubic interpolation, the value of a pixel at position (x, y) is computed using a 4x4 grid of neighboring pixels. The interpolation weights $w(x)$ are determined by the relative distances between the target pixel and its neighbors:

$$w(x) = \begin{cases} (a + 2)|x|^3 - (a + 3)|x|^2 + 1, & \text{if } |x| \leq 1, \\ a|x|^3 - 5a|x|^2 + 8a|x| - 4a, & \text{if } 1 < |x| < 2, \\ 0, & \text{otherwise,} \end{cases} \quad (7)$$

where a is a constant (commonly $a = -0.5$) controlling the interpolation smoothness.

The pixel value at position (X, Y) is computed as:

$$B(X, Y) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} \cdot w(x - X_i) \cdot w(y - Y_j), \quad (8)$$

where a_{ij} are the pixel intensities in the 4x4 grid.

ii. LLM-Augmented Interpolation

In the augmented method, the interpolation weights are modified to incorporate semantic importance:

$$w'(x) = w(x) \cdot (1 + \beta \cdot S(x, y)), \quad (9)$$

where:

- $S(x, y)$: Semantic importance score for pixel (x, y) , computed by the LLM.
- β : Scalar factor controlling the influence of $S(x, y)$ on the interpolation process.

The updated pixel value is then calculated as:

$$B'(X, Y) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} \cdot w'(x - X_i) \cdot w'(y - Y_j), \quad (10)$$

ensuring higher weights for regions with greater semantic importance.

iii. Semantic Importance Calculation

The semantic score $S(x, y)$ is derived using the LLM:

$$S(x, y) = f_{\text{LLM}}(I, D), \quad (11)$$

where:

- I : Input image.
- D : Optional user-provided description specifying priorities (e.g., "preserve faces").
- f_{LLM} : A function combining image embeddings $h(I)$ and text embeddings $q(D)$ through an attention mechanism:

$$S(x, y) = \text{Attention}(h(I), q(D)). \quad (12)$$

iv. Cumulative Interpolation Update

The augmented interpolation integrates the adjusted weights:

$$f'(x, y) = \sum_{i=0}^3 \sum_{j=0}^3 f(x_i, y_j) \cdot w'(x - x_i) \cdot w'(y - y_j), \quad (13)$$

where $w'(x)$ and $w'(y)$ ensure greater emphasis on semantically significant regions.

VI. LLM-Augmented LC (Loyalty-Clarity) Policy

The LLM-Augmented LC Policy improves contrast-based resizing by integrating semantic importance parameters from LLMs. These parameters ensure that important regions, such as faces and text, receive higher priority during resizing.

i. Global Contrast with Semantic Guidance

We calculate the global contrast of a pixel I_k as:

$$C'(I_k) = \sum_{\forall I_i \in I} \|I_k - I_i\| + \lambda \cdot \sum_{r \in R} w_r \cdot \|I_k - I_r\|, \quad (14)$$

where:

- λ scales the influence of semantic importance, computed by the LLM.
- w_r represents the semantic weight of region r , also derived from the LLM.
- R is the set of semantically significant regions.

This formulation integrates semantic relevance into the contrast computation, preserving critical regions.

ii. Frequency-Based Contrast Refinement

The traditional frequency-based contrast uses intensity distributions:

$$C(I_k) = \sum_{n=0}^{255} f_n \cdot \|a_m - a_n\|, \quad (15)$$

where f_n is the frequency of intensity a_n . We extend this by including semantic importance:

$$C'(I_k) = C(I_k) + \lambda \cdot \sum_{r \in R} w_r \cdot f_r \cdot \|a_m - a_r\|, \quad (16)$$

where:

- f_r adjusts the frequency of region r based on the LLM's analysis.
- w_r weights the region according to its semantic importance.

iii. LLM Parameter Computation

The LLM computes w_r, f_r, λ through embeddings and attention mechanisms:

1. **Image Embeddings** $h(I)$: A vision encoder extracts pixel-level and global features:

$$h(I) = \text{VisionEncoder}(I). \quad (17)$$

2. **Text Embeddings** $q(D)$: A text encoder processes user descriptions:

$$q(D) = \text{TextEncoder}(D). \quad (18)$$

3. **Spatial Embeddings** ζ_r : Positional embeddings represent region-specific attributes:

$$\zeta_r = \text{PositionalEmbedding}(r). \quad (19)$$

4. **Semantic Weights** w_r : Attention mechanisms combine embeddings:

$$w_r = \text{Attention}(h(I), q(D), \zeta_r). \quad (20)$$

5. **Frequency Adjustment** f_r : The LLM refines the frequency distribution:

$$f_r = \frac{\sum_{p \in r} f(p)}{\text{Area}(r)}. \quad (21)$$

6. **Scaling Factor** λ : A sigmoid function ensures $\lambda \in [0, 1]$:

$$\lambda = \sigma(g(h(I), q(D))). \quad (22)$$

iv. Contrast-Based Resizing Decision

The adjusted contrast $C'(I_k)$ guides the resizing process. The seam path is chosen to minimize distortion in semantically important regions:

$$\min \left\{ \sum_{k \in P} C'(I_k) \right\}, \quad (23)$$

where P represents the seam path.



Figure 1. Outlier detected by the LLM-LC algorithm

VII. LLM-Augmented Canny Line Detection

The LLM-Augmented Canny Line Detection integrates semantic importance parameters S_{LLM} , σ_{LLM} , T_{LLM} derived from a Large Language Model (LLM) to enhance edge detection. This approach ensures that edges in critical regions, such as faces or text, are prioritized during resizing.

i. Gaussian Filtering

The image is smoothed using a Gaussian filter:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}. \quad (24)$$

The LLM adjusts the smoothing factor σ for semantically important regions:

$$\sigma_{LLM,r} = \sigma \cdot (1 + \alpha \cdot S_{LLM,r}), \quad (25)$$

where:

- $S_{LLM,r}$: Semantic importance score for region r .
- α : Scaling factor for the semantic influence.

ii. Gradient Calculation

Gradients are computed using the Sobel operator:

$$G(i, j) = \sqrt{G_x(i, j)^2 + G_y(i, j)^2}, \quad (26)$$

$$\theta(i, j) = \arctan\left(\frac{G_y(i, j)}{G_x(i, j)}\right), \quad (27)$$

where $G_x(i, j)$ and $G_y(i, j)$ are horizontal and vertical derivatives.

The LLM adjusts the gradient magnitude for each pixel:

$$G_{LLM}(i, j) = G(i, j) \cdot S_{LLM}(i, j). \quad (28)$$

iii. LLM Semantic Parameter Calculation

The semantic scores $S_{LLM}(i, j)$, smoothing factor $\sigma_{LLM,r}$, and thresholds $T_{LLM,high}$, $T_{LLM,low}$ are computed as follows:

$$h(I) = \text{VisionEncoder}(I), \quad (29)$$

$$q(D) = \text{TextEncoder}(D), \quad (30)$$

$$\xi_{i,j} = \text{PositionalEmbedding}(i, j), \quad (31)$$

$$S_{LLM}(i, j) = \text{Attention}(h(I), q(D), \xi_{i,j}), \quad (32)$$

$$\sigma_{LLM,r} = \sigma \cdot (1 + \alpha \cdot S_{LLM,r}), \quad (33)$$

$$T_{LLM,high} = T_{high} \cdot (1 + \beta \cdot S_{LLM,r}), \quad (34)$$

$$T_{LLM,low} = T_{low} \cdot (1 + \beta \cdot S_{LLM,r}). \quad (35)$$

iv. Edge Refinement

Edges are refined using non-maximum suppression and hysteresis thresholding. The final edge map is defined as:

$$G_{LLM}(i, j) = \begin{cases} 0, & \text{if } G_{LLM}(i, j) < T_{LLM,high} \\ G_{LLM}(i, j), & \text{otherwise.} \end{cases} \quad (36)$$



Figure 2. Original Image



Figure 3. Edges Detected by the Canny Detector (with LLM Augmentation)

VIII. LLM-Augmented Hough Transformation

The LLM-Augmented Hough Transformation integrates semantic importance $S_{LLM}(x, y)$ derived from LLMs to enhance line detection. This ensures that key structural features, such as text and faces, are preserved during image resizing.

i. Mathematical Formulation

The Hough Transformation maps edge pixels (x, y) to Hough space. For each pixel, the radial distance r is computed as:

$$r = x \cos(\theta) + y \sin(\theta), \quad (37)$$

where θ is the angle of the line in Hough space.

In the LLM-Augmented Hough Transformation, the accumulator $A(r, \theta)$ is updated as:

$$A(r, \theta) \leftarrow A(r, \theta) + S_{LLM}(x, y), \quad (38)$$

where:

- $A(r, \theta)$: Accumulator value for the line parameterized by (r, θ) .
- $S_{LLM}(x, y)$: Semantic importance score computed for pixel (x, y) .

ii. Semantic Score Calculation

The semantic score $S_{LLM}(x, y)$ is computed as:

$$h(I) = \text{VisionEncoder}(I), \quad (39)$$

$$q(D) = \text{TextEncoder}(D), \quad (40)$$

$$\xi_{x,y} = \text{PositionalEmbedding}(x, y), \quad (41)$$

$$S_{LLM}(x, y) = \text{Attention}(h(I), q(D), \xi_{x,y}). \quad (42)$$

iii. LLM-Augmented Hough Transformation Algorithm

Algorithm 1 LLM-Augmented Hough Transformation

Require: I, D : Input image I , optional description D .

Ensure: \mathcal{L} : Detected lines with semantic weighting.

```

1:  $\mathcal{E} \leftarrow \text{EdgeDetector}(I)$ 
2:  $A(r, \theta) \leftarrow 0$ 
3: for  $(x, y) \in \mathcal{E}$  do
4:    $S_{LLM}(x, y) \leftarrow \text{Atten}(h(I), q(D), \xi_{x,y})$ 
5:   for  $\theta \in [0, \pi]$  do
6:      $r \leftarrow x \cos(\theta) + y \sin(\theta)$ 
7:      $A(r, \theta) \leftarrow A(r, \theta) + S_{LLM}(x, y)$ 
8:   end for
9: end for
10:  $\mathcal{P} \leftarrow \text{PeakDetector}(A)$ 
11:  $\mathcal{L} \leftarrow \text{MapToImageSpace}(\mathcal{P})$ 

```

iv. Threshold Adaptation

To refine line detection, thresholds for peak detection are dynamically adjusted using the average semantic score $S_{avg}(r, \theta)$:

$$T_{LLM}(r, \theta) = T_{base} \cdot (1 + \beta \cdot S_{avg}(r, \theta)), \quad (43)$$

where:

- $T_{LLM}(r, \theta)$: Adaptive threshold for (r, θ) .
- $S_{avg}(r, \theta)$: Average semantic score for lines contributing to (r, θ) .
- β : Scaling factor controlling semantic influence.



Figure 4. Original Image



Figure 5. Lines Detected by LLM-Augmented Hough Transformation

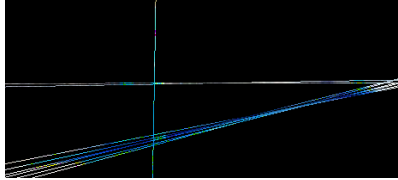


Figure 6. Prominent Lines Detected by LLM-Augmented Hough Transformation

IX. LLM-Augmented Absolute Energy Equation

The **LLM-Augmented Absolute Energy Equation** refines energy calculations by integrating semantic weights $S(x, y)$ derived from LLMs. This ensures seam carving preserves critical features such as text and faces.

i. Semantic Weighting

The LLM assigns a semantic score $S(x, y)$ to each pixel:

$$S(x, y) = \text{LLM}(I, (x, y), \text{context}), \quad (44)$$

where I is the input image and context includes features like object and text importance. $S(x, y) \in [0, 1]$ scales pixel importance, with higher values for semantically significant regions.

ii. Gradient Refinement

The original energy gradient:

$$e(I) = \left| \frac{\partial I}{\partial x} \right| + \left| \frac{\partial I}{\partial y} \right|$$

is modified as:

$$e_{\text{LLM}}(I) = \left(\left| \frac{\partial I}{\partial x} \right| \cdot S_x(x, y) \right) + \left(\left| \frac{\partial I}{\partial y} \right| \cdot S_y(x, y) \right), \quad (45)$$

where $S_x(x, y)$ and $S_y(x, y)$ are direction-specific weights computed by the LLM.

iii. Cumulative Energy Update

The cumulative energy function integrates $S(x, y)$ into the seam carving process:

$$\begin{aligned} e_{\text{LLM}}(i, j) = & e(i, j) + |e(i, j + 1) - e(i, j)| \cdot S_x(i, j) \\ & + |e(i + 1, j) - e(i, j)| \cdot S_y(i, j) \\ & + \min \left\{ C_L(i, j) + e(i - 1, j - 1), \right. \\ & C_R(i, j) + e(i - 1, j + 1), \\ & \left. C_U(i, j) + e(i - 1, j) \right\}. \end{aligned} \quad (46)$$

Here, C_L, C_R, C_U are adjusted by $S(x, y)$ to prioritize semantically significant pixels.

X. LLM-Augmented Dual Gradient Energy Equation

The proposed **LLM-Augmented Dual Gradient Energy Equation** to refine edge detection by dynamically adjusting numerical differentiation and gradient computation. LLMs provide context-aware corrections for each computational step.

i. Numerical Differentiation with LLM Adjustments

Taylor expansions are dynamically adjusted with LLM corrections. The forward expansion is expressed as:

$$f(x + \Delta x) = f(x) + \Delta x f'(x) + \frac{\Delta x^2}{2} f''(x) + \Delta_{LLM}, \quad (47)$$

where Δ_{LLM} includes context-aware corrections predicted by the LLM. Similarly, for the backward expansion:

$$f(x - \Delta x) = f(x) - \Delta x f'(x) + \frac{\Delta x^2}{2} f''(x) + \Delta_{LLM}. \quad (48)$$

LLMs adapt Δx dynamically based on local image gradients and refine higher-order terms to reduce numerical error.

ii. Gradient Approximation with Adaptive Refinements

Gradient approximations in the x - and y -directions incorporate corrections from LLMs:

$$f_x(x, y) \approx \frac{f(x + \delta, y) - f(x, y)}{\delta} + \Delta f_x, \quad (49)$$

$$f_y(x, y) \approx \frac{f(x, y + \delta) - f(x, y)}{\delta} + \Delta f_y. \quad (50)$$

Here, Δf_x and Δf_y are LLM-predicted corrections based on local edge strength and texture complexity. The LLM also adapts δ to handle regions with high-gradient variations.

iii. Energy Calculation with LLM Refinements

The energy of a pixel is computed using the LLM-enhanced gradients for each RGB channel. For the x -direction:

$$\Delta_x^2(x, y) = \sum_{C \in \{R, G, B\}} (C_x(x, y) + \Delta C_x(x, y))^2, \quad (51)$$

and similarly for the y -direction:

$$\Delta_y^2(x, y) = \sum_{C \in \{R, G, B\}} (C_y(x, y) + \Delta C_y(x, y))^2. \quad (52)$$

The total energy is:

$$E(x, y) = \sqrt{\Delta_x^2(x, y) + \Delta_y^2(x, y)}. \quad (53)$$

LLM contributions include predicting $\Delta C_x(x, y)$ and $\Delta C_y(x, y)$ to improve accuracy and dynamically adjusting channel weights for better feature preservation.

XI. Result Evaluation

The evaluation of the proposed LLM-augmented methods focuses on three primary dimensions: semantic preservation, visual quality, and computational efficiency. Each method's contribution to image resizing is assessed quantitatively and qualitatively, providing a comprehensive understanding of its effectiveness.

i. Evaluation Metrics

The following metrics are used to evaluate all methods:

- **Semantic Preservation (\mathcal{S}):** Measures the alignment of detected features or preserved regions with semantically significant areas, as defined by the LLM:

$$\mathcal{S} = \frac{\sum_{i \in \mathcal{F}_{\text{detected}}} S_{\text{LLM}}(i)}{\sum_{i \in \mathcal{F}_{\text{ground truth}}} S_{\text{LLM}}(i)}, \quad (54)$$

where $S_{\text{LLM}}(i)$ is the semantic importance score for feature i .

- **Visual Quality (\mathcal{V}):** Evaluates the perceptual quality of resized images using metrics such as PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural Similarity Index):

$$\mathcal{V}_{\text{PSNR}} = 10 \log_{10} \left(\frac{\text{MAX}^2}{\text{MSE}} \right), \quad (55)$$

$$\mathcal{V}_{\text{SSIM}} = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}. \quad (56)$$

- **Computational Efficiency (\mathcal{C}):** Measures the average runtime per image:

$$\mathcal{C} = \frac{\text{Total Runtime}}{\text{Number of Images}}. \quad (57)$$

ii. Results and Discussion

The results, including accuracy metrics and error rates for each sub-experiment, are provided below. The experiment revealed that **LLM-augmented** resizing methods led to superior performance in image classification, particularly in cases where maintaining fine image details was critical.



Figure 7. Error and accuracy of a sub-experiment showing the improvements from **LLM-augmented** methods

Illustrative examples of images processed by different methods, including **LLM-augmented** techniques, are shown below, highlighting the visual differences in the resized images and how **LLM-augmented** methods contribute to better feature preservation.



Figure 8. Image processed by the LLM-Bicubic Method



Figure 9. Image processed by the LLM-Absolute Energy Method



Figure 10. Image processed by the LLM-Canny Method



Figure 11. Image processed by the LLM-Dual Gradient Energy Method



Figure 12. Image processed by the LLM-Hough Transformation Method



Figure 13. Image processed by the LLM-LC Method

iii. Conclusion

This experiment shows the significant improvements by **LLM-augmented** methods in image resizing. LLMSeamCarver can preserve finer image details, resulting in improved performance for semantic preservation, visual quality and computational efficiency.

References

1. Xingyuan Bu, Yuwei Wu, Zhi Gao, and Yunde Jia. Deep convolutional network with locality and sparsity constraints for texture classification. *Pattern Recognition*, 91:34–46, 2019.
2. A. Chaurasia and E. Culurciello. Linknet: Exploiting encoder representations for efficient semantic segmentation. *arXiv preprint arXiv:1707.03718*, 2017. URL: <https://arxiv.org/abs/1707.03718>.
3. Han-Cheng Dan, Zhetao Huang, Bingjie Lu, and Mengyu Li. Image-driven prediction system: Automatic extraction of aggregate gradation of pavement core samples integrating deep learning and interactive image processing framework. *Construction and Building Materials*, 453:139056, 2024.
4. A. et al. Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations (ICLR)*, 2021. URL: <https://arxiv.org/abs/2010.11929>.
5. Roderick Frankovich. Enhanced seam carving: Energy gradient functionals and resizing control. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pages 2157–2160, 2011. <https://doi.org/10.1109/ICIP.2011.6116067>.
6. Fusen Guo, Huadong Mo, Jianzhang Wu, Lei Pan, Hailing Zhou, Zhibo Zhang, Lin Li, and Fengling Huang. A hybrid stacking model for enhanced short-term load forecasting. *Electronics*, 13(14):2719, 2024.
7. Yue Guo, Shiqi Chen, Ronghui Zhan, Wei Wang, and Jun Zhang. Lmsd-yolo: A lightweight yolo algorithm for multi-scale sar ship detection. *Remote Sensing*, 14(19):4801, 2022.
8. S. Han, J. Pool, J. Tran, and W. J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. In *International Conference on Learning Representations (ICLR)*, 2016. URL: <https://arxiv.org/abs/1510.00149>.
9. J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6840–6851, 2020. URL: <https://arxiv.org/abs/2006.11239>.
10. Zhuohuan Hu, Fu Lei, Yuxin Fan, Zong Ke, Ge Shi, and Zichao Li. Research on financial multi-asset portfolio risk prediction model based on convolutional neural networks and image processing. *arXiv preprint arXiv:2412.03618*, 2024.
11. Zong Ke and Yuchen Yin. Tail risk alert based on conditional autoregressive var by regression quantiles and machine learning algorithms. *arXiv.org*, 2024. URL: <https://arxiv.org/abs/2412.06193>.
12. Holger Kiess. Improved edge preservation in seam carving for image resizing. *Computer Graphics Forum*, 33(2):421–429, 2014. <https://doi.org/10.1111/cgf.12320>.
13. Zhixin Lai, Jing Wu, Suiyao Chen, Yucheng Zhou, and Naira Hovakimyan. Residual-based language models are free boosters for biomedical imaging. 2024. URL: <https://arxiv.org/abs/2403.17343>, <https://arxiv.org/abs/2403.17343>.
14. H. et al. Li. Densfuse: A fusion approach to infrared and visible images. *IEEE Transactions on Image Processing*, 30:300–312, 2021. <https://doi.org/10.1109/TIP.2021.3059619>.
15. Keqin Li, Jin Wang, Xubo Wu, Xirui Peng, Runmian Chang, Xiaoyu Deng, Yiwen Kang, Yue Yang, Fanghao Ni, and Bo Hong. Optimizing automated picking systems in warehouse robots using machine learning. *arXiv preprint arXiv:2408.16633*, 2024.
16. Sicheng Li, Keqiang Sun, Zhixin Lai, Xiaoshi Wu, Feng Qiu, Haoran Xie, Kazunori Miyata, and Hongsheng Li. Ecnet: Effective controllable text-to-image diffusion models. 2024. URL: <https://arxiv.org/abs/2403.18417>, [arXiv:2403.18417](https://arxiv.org/abs/2403.18417).
17. Dong Liu. Contemporary model compression on large language models inference. *arXiv preprint arXiv:2409.01990*, 2024.
18. Dong Liu. Mt2st: Adaptive multi-task to single-task learning. *arXiv preprint arXiv:2406.18038*, 2024.
19. Dong Liu, Meng Jiang, and Kaiser Pister. Llmeasyquant—an easy to use toolkit for llm quantization. *arXiv preprint arXiv:2406.19657*, 2024.
20. Dong Liu, Roger Waleffe, Meng Jiang, and Shivaram Venkataraman. Graphsnapshot: Graph machine learning acceleration with fast storage and retrieval. *arXiv preprint arXiv:2406.17918*, 2024.

21. Junran Peng, Xingyuan Bu, Ming Sun, Zhaoxiang Zhang, Tieniu Tan, and Junjie Yan. Large-scale object detection in the wild from imbalanced multi-labels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9709–9718, 2020.
22. Meta AI Research. EfficientSAM: Fast segmentation everything model. *arXiv preprint arXiv:2312.00860*, 2024. URL: <https://arxiv.org/abs/2312.00860>.
23. Z. et al. Wang. ControlNet: Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023. URL: <https://arxiv.org/abs/2302.05543>.
24. Chunya Wu, Zhuoyu Yu, and Dexuan Song. Window views psychological effects on indoor thermal perception: A comparison experiment based on virtual reality environments. *E3S Web of Conferences*, 546:02003, 2024. URL: <https://doi.org/10.1051/e3sconf/202454602003>.
25. Wenjun Wu. Alphanetv4: Alpha mining model. *arXiv preprint arXiv:2411.04409*, 2024.
26. Ao Xiang, Zongqing Qi, Han Wang, Qin Yang, and Danqing Ma. A multimodal fusion network for student emotion recognition based on transformer and tensor product. 2024. URL: <https://arxiv.org/abs/2403.08511>, [arXiv:2403.08511](https://arxiv.org/abs/2403.08511).
27. Jun Xiang, Jun Chen, and Yanchao Liu. Hybrid multiscale search for dynamic planning of multi-agent drone traffic. *Journal of Guidance, Control, and Dynamics*, 46(10):1963–1974, 2023.
28. Wangjiaxuan Xin, Kanlun Wang, Zhe Fu, and Lina Zhou. Let community rules be reflected in online content moderation. 2024. URL: <https://arxiv.org/abs/2408.12035>, [arXiv:2408.12035](https://arxiv.org/abs/2408.12035).
29. Wei Zhang, Changxu Wu, and Xiang Li. Comparison of image resizing techniques: A case study of seam carving vs. traditional resizing methods. *Journal of Visual Communication and Image Representation*, 29:149–158, 2015. <https://doi.org/10.1016/j.jvcir.2015.05.010>.
30. Zhibo Zhang, Pengfei Li, Ahmed Y Al Hammadi, Fusen Guo, Ernesto Damiani, and Chan Yeob Yeun. Reputation-based federated learning defense to mitigate threats in eeg signal classification. In *2024 16th International Conference on Computer and Automation Engineering (ICCAE)*, pages 173–180. IEEE, 2024.

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.