A Visual Complexity Measurement Method Based on Monte Carlo Sampling

Sihan Wang(ryoya.hh@gmail.com)
School of Artificial Intelligence and Computer Science, Jiangnan University
Wuxi, China

Ruimin Lyu* (Corresponding author, ruiminlyu@jiangnan.edu.cn)
School of Artificial Intelligence and Computer Science, Jiangnan University
Wuxi, China

Abstract

Conventional visual complexity measurement faces challenges in efficiency, accuracy, and alignment with human perception. To address these, this paper presents a novel Monte Carlo-based method for visual complexity measurement, the random line segment width sampling algorithm (RLSWSA). RLSWSA employs local stochastic sampling for efficient estimation of symbol perimeter complexity. By discarding global scanning in favor of local sampling, RLSWSA significantly enhances computational efficiency while maintaining high accuracy and robustness. Experimental results show rapid convergence with just 24 samples, yielding high consistency with traditional methods (correlation coefficient > 0.9). Furthermore, RLSWSA's Spearman correlation with human subjective ratings is 0.74, demonstrating its strong correlation with human perceptual complexity. This study offers an efficient and reliable solution for rapid symbol visual complexity calculation, with strong potential for applications like symbol recognition and design optimization.

Keywords: Visual Complexity; Monte Carlo Sampling; Symbolic Measurement; Computational Efficiency; Subjective Consistency

Introduction

Visual complexity, as a core indicator of graphic cognition, holds dual significance in understanding human perceptual mechanisms and optimizing visual design. In psychology, Donderi (2006)'s systematic review reveals deep connections between visual complexity and cognitive load, attention allocation, and memory retention. In computer vision, visual complexity measurement has become a key preprocessing module for tasks such as image retrieval Rigau, Feixas, and Sbert (2005) and target detection Nagle and Lavie (2020). Research in art design focuses on balancing complexity and user experience; Tuch, Bargas-Avila, Opwis, and Wilhelm (2009) found through eye-tracking experiments that webpage visual complexity exhibits an inverted U-shaped relationship with user dwell time. In typography, Chang, Chen, and Perfetti (2018)'s multidimensional analysis of 131 writing systems indicates that perimeter complexity can account for 76% of cross-cultural design differences.

Despite the widespread application of visual complexity measurement methods across various fields, existing research has paid insufficient attention to the "efficiency-accuracy-subjective consistency" balance, and their limitations continue to restrict practical applications. Traditional geometric methods (e.g., Watson (2011)'s perimeter expectation model) can precisely calculate the complexity of binary images; however, their time complexity is as high as $O(n^2)$, making them unsuitable for processing large-scale data. Machine learning methods (e.g., the GraphCom multidimensional framework proposed by Chang et al. (2018)) have improved perceptual consistency, yet they rely on annotated data and lack interpretability. Dai et al. (2022)'s hierarchical perceptual model, although achieving a 92% fit to human ratings, still incurs computational costs that limit its practical application.

To address these issues, this study proposes a Monte Carlo sampling-based random line segment width sampling algorithm (RLSWSA). Compared with traditional complexity measurement methods, RLSWSA collects stroke width data of symbols by randomly generating line segments to rapidly estimate the symbol's perimeter and overall complexity, thereby effectively balancing computational efficiency and accuracy. Experimental results demonstrate that the algorithm exhibits high accuracy in computing multiple complexity metrics, and its measurements are highly consistent with human subjective perceptions of complexity.

The main contributions of this paper are as follows:

1. An efficient symbolic visual complexity measurement algorithm, RLSWSA, is proposed. Based on Monte Carlo local sampling, it replaces global contour tracing with random line segment sampling to probabilistically estimate the symbol's perimeter complexity.

2. The algorithm's ability to balance efficiency and accuracy is demonstrated. With a limited number of samples, RLSWSA achieves high-precision computational results comparable to traditional methods, effectively overcoming the high computational resource bottleneck of conventional approaches.

3. A strong correlation ($r = 0.74$) between perimeter complexity and human visual perception is validated through experiments, providing a quantifiable indicator of perceptual consistency for symbol design optimization.

Related Work

Visual Complexity Measurement Methods

Research on visual complexity measurement methods mainly focuses on the following aspects:

In D. Barner, N.R. Bramley, A. Ruggeri and C.M. Walker (Eds.), *Proceedings of the 47th Annual Conference of the Cognitive Science Society*

Geometry-based Complexity Measurement Quantitative analysis of geometric shapes provides a foundational framework for visual complexity research. Attneave and Arnoult (1956) were the first to link topological features with cognitive load. Majaj, Pelli, Kurshan, and Palomares (2002) proposed the spatial frequency channel theory, which predicts letter recognition efficiency based on stroke frequency. Pelli, Burns, Farell, and Moore-Page (2006) further introduced Perimetric Complexity and demonstrated its significant correlation with letter recognition response time. However, traditional methods rely on global contour tracing, which results in high computational complexity, limiting the efficiency of processing large-sized images.

Perception-based Complexity Measurement To capture human subjective perception, researchers have developed various psychophysical measurement paradigms. Purchase, Freeman, and Hamer (2012) conducted subjective rating experiments with nine categories of images, confirming that features such as symmetry and color diversity influence complexity perception. Guo, Qian, Li, and Asano (2018) constructed a regression model with 29 global/local features, achieving an 86.78% classification accuracy on a painting dataset. However, existing methods generally suffer from high experimental costs, poor cross-domain generalization, and a lack of explicit connections between perceptual features and geometric metrics.

Deep Learning-based Complexity Measurement In recent years, deep neural networks have provided a new paradigm for complexity modeling. Nagle and Lavie (2020) used a pre-trained VGG16 network to predict scene complexity, achieving a correlation coefficient of 0.83, outperforming traditional metrics such as JPEG compression ratios. Saraee, Jalal, and Betke (2020) proposed a complexity measurement method based on intermediate layer activation energy, explaining 51% of aesthetic differences in an advertisement image dataset. Although deep learning methods demonstrate powerful feature learning capabilities, their black-box nature leads to insufficient interpretability, and their reliance on large-scale annotated data limits their applicability in low-resource scenarios.

Perimetric Complexity

Perimetric complexity serves as a bridge between geometric features and perceptual evaluation, aiming to measure the complexity of a symbol's boundary. The index is calculated based on the foreground perimeter P and the area A of the symbol image, defined as follows:

$$C = \frac{P^2}{4\pi A}$$

Where C represents the perimetric complexity, P denotes the symbol's perimeter, and A is the symbol's area. Experimental studies have shown that the correlation coefficient between perimetric complexity and human subjective ratings can reach up to 0.74 , making it the strongest single-factor indicator for predicting perceived visual complexity.

Traditional perimeter calculations rely on contour tracing algorithms. For example, the Pelli algorithm employs thinning processes to obtain a single-pixel-wide contour, with a time complexity of $O(W^2)$ (where W is the image width). Although Watson (2011)'s improved boundary scanning method enhances accuracy, its computational complexity still increases quadratically with image resolution.

The proposed RLSWSA algorithm overcomes this bottleneck by adopting a Monte Carlo sampling strategy, reducing the complexity to $O(S \cdot W)$ (where S is the number of samples). This approach balances efficiency, accuracy, and subjective consistency, thereby significantly addressing the limitations of existing methods.

Approach

RLSWSA Algorithm

In this study, we propose a Monte Carlo-based random line segment width sampling algorithm (RLSWSA) for estimating the distribution of stroke widths in symbol images to approximate the perimetric complexity of the image.

Single Random Sampling The process for approximating stroke width is as follows:

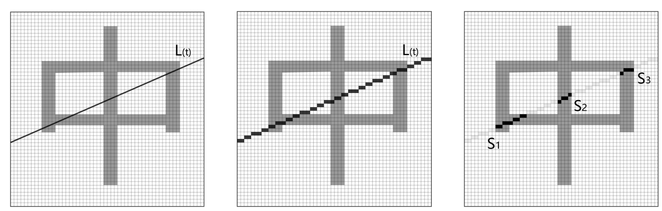

Figure 1: RLSWSA steps: (a) Line generation, (b) Pixel traversal, (c) Segment identification.

1. Random Line Generation: As shown in Figure 1(a), on the symbol image plane Ω, a parameterized line $L(t) = P_0 + t \cdot v$ is randomly generated, where the initial point $P_0 \in \Omega$ and the direction vector $v \in (0, 2\pi)$. This ensures uniformity and diversity in sampling, thereby covering different regions of the image.
2. Pixel Traversal Analysis: As shown in Figure 1(b), the Bresenham algorithm is used along the line $L(t)$ to extract pixel data, resulting in the pixel sequence $(p_1, p_2, \dots, p_n)$ covered by $L(t)$. This algorithm efficiently determines all the pixels on the line using integer steps, avoiding the performance overhead associated with floating-point operations while ensuring precise sampling.
3. Effective Line Segment Identification: Define a pixel criterion $f(p_i) = 1$ (foreground) or 0 (background). A sliding window is used to detect consecutive foreground pixel segments, which are regarded as the symbol's stroke segments,

denoted as $s_k = [p_i, \ldots, p_j]$. As shown in Figure 1(c), assume that m effective segments $\{s_1, \ldots, s_m\}$ are obtained in a single experiment. Then, the local width of each segment can be expressed as:

$$W_k = |s_k| \times \Delta s$$

where $|s_k|$ is the number of consecutive foreground pixels, and $\Delta s = \sqrt{1 + \frac{dx^2}{dy^2}}$ is the geometric correction factor for the unit step.

Monte Carlo Algorithm for Calculating the Average Stroke Width By conducting S independent experiments, we obtain the sampling distribution $\{W_s\}$, which reflects the overall characteristics of stroke widths in the symbol image. The average stroke width of the symbol is obtained by calculating the mean of $\{W_s\}$, effectively overcoming the issue of non-uniformity in edge complexity. The average stroke width can converge to:

$$\mu_W = \frac{1}{S} \sum_{s=1}^{S} \left[\frac{1}{m_s} \sum_{k=1}^{m_s} W_{s,k} \right]$$

Visual Complexity Metrics To comprehensively assess the visual complexity of symbol images, this study employs two complexity metrics: line length and perimetric complexity, defined as follows:

1. Line Length: The overall line length of the symbol is measured by calculating the total number of foreground pixels in the image and dividing it by the average stroke width:

   $$L = \frac{A}{\mu_W}$$

   where $A = |\{p \mid f(p) = 1\}|$ represents the total number of foreground pixels.

2. Perimetric Complexity: In this algorithm, the equivalent foreground perimeter model is defined as

   $$P = 2\left(L + A \cdot \mu_W\right).$$

   Substituting this into the classical perimetric complexity formula,

   $$C = \frac{P^2}{4\pi A},$$

   we obtain the optimized expression:

   $$C = \frac{\left[2\left(L + A \cdot \mu_W\right)\right]^2}{4\pi A} = \frac{\left(L + A \cdot \mu_W\right)^2}{\pi A}.$$

   This definition significantly reduces the computational overhead of boundary tracing while retaining the traditional geometric meaning of the metric.

Comparison of Time Complexity with the Pelli Algorithm

The Pelli algorithm, originally proposed by Pelli et al., is a widely used method for estimating perimetric complexity in binary images. This algorithm operates by first converting the original image into a one-pixel-wide contour line. It then processes this contour to approximate the perimeter of the symbol. The Pelli algorithm is known for its ability to quickly estimate the symbol's boundary length without the need for complex curve tracing, thus providing a foundation for calculating perimetric complexity. Its implementation is relatively straightforward, offers good accuracy, and is suitable for processing medium to small-scale binary symbol images. However, its core operation involves a global traversal of the image pixels, leading to a time complexity of $O(W^2)$ for an image with width W.

In contrast, the proposed Monte Carlo sampling-based RLSWSA algorithm estimates stroke widths through a random line sampling approach. The fundamental difference in processing methodology results in a significant difference in time complexity, as shown in Table 1:

Table 1: Comparison of Time Complexity Between Algorithms.

Algorithm	Time Complexity	Description
Pelli	$O(W^2)$	Global pixel traversal
RLSWSA	$O(S \cdot W)$	Random line sampling

- **Efficiency Advantage:** When the number of samples S is considerably smaller than the image width W, the time complexity of RLSWSA, $O(S \cdot W)$, is substantially lower than that of the Pelli algorithm, $O(W^2)$. For example, with an image width of $W = 1000$ pixels and $S = 100$ samples, the time complexity of Pelli would be 10^6, while RLSWSA's would be 10^5, roughly one-tenth of Pelli's complexity.
- **Accuracy and Efficiency Trade-off:** RLSWSA offers a balance between computational efficiency and estimation accuracy by controlling the number of samples S. Appropriately selecting S allows for achieving sufficient complexity estimation accuracy while maintaining high efficiency.

Datasets and Experimental Design

Symbol Dataset Composition

To comprehensively validate the effectiveness of the RLSWSA algorithm in measuring symbol visual complexity, we constructed a multilingual symbol dataset. This dataset comprises 940 characters covering 12 languages and writing systems, aiming to reflect the visual diversity of global symbols. All symbols were uniformly rendered as 1024×1024 pixel binary images for algorithm input. The specific composition is presented in the Table 2 below:

Table 2: Composition of the 940C Multilingual Symbol Dataset.

Character Type	Quantity	Symbol Examples
Latin Alphabet	51	A, B, C, D, ..., a, b, c, d, ...
Greek Alphabet	48	Α, Β, Γ, Δ, ..., α, β, γ, δ, ...
Cyrillic Alphabet	66	Ё, А, Б, В, ..., а, б, в, г, ...
Armenian Alphabet	72	Ա, Բ, Գ, Դ, ..., ա, բ, գ, դ, ...
Hebrew Alphabet	23	א, ב, ג, ד, ...
Arabic Alphabet	117	أ, ؤ, ئ, ء, ...
Indic Writing Systems	280	अ, आ, ⋯, অ, আ, ⋯, ਅ, ਕ, ⋯, அ, ஆ, ⋯, అ, ఆ, ⋯, క, వ, ⋯
Thai Script	44	ก, ข, ฃ, ค, ฅ, ...
Georgian Alphabet	33	ა, ბ, გ, დ, ⋯
Japanese Characters	45	あ, い, う, え, お, ...
Korean Hangul	31	ㄱ, ㄲ, ㄴ, ㄷ, ㄸ, ...
Chinese Characters	130	丁, 上, ..., 張, 愛, ...

Subjective Complexity Evaluation Experiment

To obtain human subjective complexity ratings for comparison with algorithmically computed results, we conducted a subjective evaluation experiment using the aforementioned dataset. The experiment was conducted in a controlled laboratory environment.

Figure 2: Experimental interface for subjective complexity rating.

Experimental Design and Procedure

1. **Participants**: A total of 202 participants (aged 18-26) with normal or corrected-to-normal vision were recruited. All participants passed a visual acuity test. The experiment adhered to ethical guidelines and obtained informed consent.

2. **Stimulus Presentation**: As illustrated in Figure 2, symbols were presented uniformly at the center of a pure white background.

3. **Rating Task**: Participants used a continuous 0-100 slider, labeled "simple" and "complex" at the ends, to directly evaluate the visual complexity of each character. Each participant completed 5 trials, with each trial containing 21 randomly presented characters. To monitor attention, the first character in each trial had significantly lower complexity than the last character.

Data Quality Control and Processing A total of 1010 trial data points were initially collected. To ensure data quality, the following control measures were applied:

1. **Attention Check**: Trials with abnormal ratings for the attention check characters (first and last) were excluded.

2. **Response Time Filtering**: Trials completed in less than 30 seconds were excluded to filter out inattentive responses. This threshold was empirically determined to ensure sufficient time was spent on each character rating.

After quality control, 1005 trials were deemed valid. The raw ratings within valid trials were normalized using min-max scaling to the range [0, 1]. The final subjective rating for each character was computed as the weighted average of its normalized ratings across all valid trials. We performed Spearman correlation analysis between the RLSWSA computed results and the final subjective ratings, and reported the significance test results.

Experiments and Results

This chapter presents experimental results evaluating the proposed RLSWSA method for symbolic visual complexity measurement, focusing on rapid convergence and subjective consistency

Rapid Convergence and Accuracy Validation

This study evaluated RLSWSA for rapid convergence and computational accuracy, with a focus on analyzing the impact of different numbers of random samples on the error, runtime, and the accuracy difference compared to traditional complexity calculation methods.

Error Analysis and Convergence Analysis The convergence of RLSWSA at different number of samples S was quantified using Percentage Error, defined against the traditional Pelli algorithm's results (C_{Pelli}). For each symbol, RLSWSA's calculated perimetric complexity at S is $C_{\text{RLSWSA},S}$. The Percentage Error is calculated as:

$$\text{Error}_S = \frac{|C_{\text{RLSWSA},S} \times \alpha - C_{\text{Pelli}}|}{C_{\text{Pelli}}} \times 100\%$$

where $\alpha = 1.618$ is an empirically determined adjustment coefficient used to calibrate the original calculation results of RLSWSA, bringing its numerical range closer to that of the Pelli algorithm for a fairer assessment of relative error.

Figure 3 illustrates the Percentage Error distribution across 940 symbols for RLSWSA compared to the Pelli algorithm at varying S (1-256). Thin lines show individual symbol error trajectories, while the thick red line represents the average error. The figure clearly shows that as S increases, errors

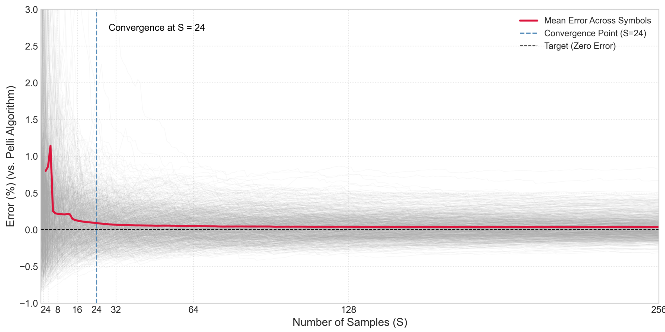

Figure 3: Percentage error distribution of RLSWSA compared to the Pelli algorithm for perimetric complexity calculation at different sampling counts. Individual symbol error trajectories are shown by thin lines, and the average error by the thick red line. The convergence point is indicated at S=24.

generally decrease, error dispersion narrows, and the results stabilize, approaching the Pelli algorithm's benchmark. The average error also declines rapidly and stabilizes.

To more precisely determine the convergence point of the algorithm, we analyzed the rate of change of the average Percentage Error. The algorithm is considered to have reached robust convergence when the absolute change in the average error within 5 consecutive sampling increment windows (e.g., from $S=1$ to $S=5$, $S=2$ to $S=6$, ...) is less than a preset threshold of 0.005. The analysis results show that when the number of samples $S=24$, the change in the average error of the RLSWSA algorithm is already very small, reaching a state of robust convergence. This indicates that the algorithm can obtain stable calculation results with a relatively small number of samples.

Computational Efficiency RLSWSA algorithm not only demonstrates good convergence but also has significant advantages in computational efficiency. We conducted runtime tests under the following Experimental Hardware Environment: processor 13th Gen Intel(R) Core(TM) i5-13400, memory 32GB DDR4 2666MHz.

Table 3 shows the total runtime required by the RLSWSA algorithm and the Pelli algorithm to process all 940 symbols at different number of samples S.

Table 3: Runtime Comparison Between Algorithms.

Algorithm	Number of Samples	Time (s)
RLSWSA	16	6.269
	24	6.289
	32	6.322
	64	6.556
	128	7.078
	256	7.764
Pelli	-	93.006

From the data in the table, it can be seen that the runtime of the RLSWSA algorithm shows an approximately linear relationship with the number of samples S. For example, when $S=24$ reaches robust convergence, the total time to process all 940 symbols is approximately 6.289 seconds. This means that the average computation time per symbol is less than 0.01 seconds (6.289 / 940 $\approx$ 0.0066 seconds). Considering that the symbol image size is 1024x1024 pixels, the Time Complexity of the traditional Pelli algorithm is $O(W^2)$, where W is the image width (1024). Its computation volume is far greater than RLSWSA's $O(SW)$ (for $S=24$). Therefore, the RLSWSA algorithm has an overwhelming efficiency advantage compared to the quadratic complexity of traditional methods, especially when processing large-scale datasets.

Accuracy Performance To comprehensively evaluate the Accuracy of RLSWSA, we analyzed its consistency with traditional methods and human subjective perception. First, we compared the calculation results of RLSWSA with traditional methods. As shown in Figure 4, at $S=24$, the two already show a high degree of correlation. As the number of samples increases, the Spearman correlation coefficient between RLSWSA and traditional methods can reach above 0.9, indicating a significant linear relationship between the two.

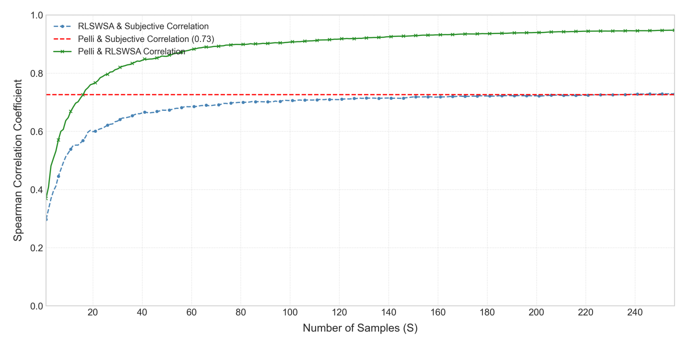

Figure 4: Correlation between RLSWSA and Traditional Algorithm Complexity Measurements at Different Sampling Counts.

Through Accuracy validation, RLSWSA demonstrates its ability to yield precise outcomes close to traditional methods under high sampling conditions, while maintaining high efficiency computation under low sampling conditions. Its rapid convergence characteristics and high degree of consistency with traditional methods make RLSWSA a practical and effective Tool for symbolic complexity measurement in real-world scenarios.

Subjective Consistency

In addition to evaluating the objective performance of the algorithm, this study further validated the subjective consistency between the visual complexity calculated by RLSWSA and human subjective Perception. We analyzed the correlation coefficient between the perimetric complexity calculated by RLSWSA and the traditional Pelli algorithm with the subjective complexity ratings dataset of 940 symbols.

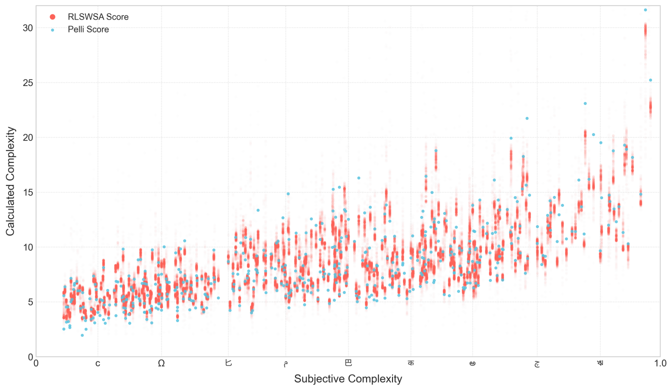

Figure 5: Scatter plots of algorithm complexity measurements (RLSWSA: red, Traditional: blue) versus human subjective ratings. Red point transparency indicates the number of RLSWSA samples.

Figure 5 presents scatter plots comparing RLSWSA and traditional Pelli algorithm complexity measurements with human subjective ratings. The vertical axis represents the subjective complexity ratings. As depicted, the distribution of RLSWSA results (red points) shows a clear dependency on the number of samples, indicated by transparency. With increasing sample counts (red points become less transparent), the RLSWSA scatter distribution progressively stabilizes and aligns more closely with the distribution of traditional method results (blue points). Both algorithms exhibit a discernible linear relationship with the human subjective ratings. This visual analysis suggests that as the number of samples for RLSWSA increases, its complexity estimations not only converge towards traditional methods but also demonstrate a more stable and consistent relationship with perceived human complexity.

Through Spearman correlation coefficient analysis, we found a highly significant correlation between the perimetric complexity calculated by RLSWSA and human subjective ratings (Spearman r = 0.74, $p < 0.001$). The traditional Pelli algorithm yielded a comparable correlation (Spearman r = 0.73, $p < 0.001$). While these strong correlations validate that perimetric complexity is a key factor, the imperfect correlation suggests that human perception is also influenced by other factors like symmetry, regularity, and feature density, not fully captured by this single metric. Future work will explore integrating these dimensions. The consistency between RLSWSA's results and subjective ratings, comparable to traditional methods, validates RLSWSA as an efficient and accurate objective measurement tool that reflects human perception.

Conclusion

This study proposes the random line segment width sampling algorithm(RLSWSA), an efficient and interpretable Monte Carlo-based method for symbolic visual complexity measurement. By employing local stochastic sampling, RLSWSA overcomes the computational limitations of traditional geometric methods while maintaining high accuracy and strong consistency with human subjective perception. This approach offers a valuable and rapidly computable metric, perimetric complexity, which can serve as an effective solution for rapid symbol complexity analysis and potentially as a useful feature for data-driven tasks like machine learning, particularly in data-constrained or real-time scenarios.

Specifically, the main contributions and findings include:

- **Innovative Algorithm Design:** Introduced a Monte Carlo sampling-based RLSWSA, which estimates stroke width and perimetric complexity by randomly generating line segments. This method replaces the high computational cost of global scanning in traditional approaches with Efficient Approximation through local sampling, significantly enhancing computational efficiency while maintaining high accuracy and Robustness for complex symbol structures.

- **Efficient and Accurate Complexity Measurement:** Experiments confirm the algorithm's rapid convergence: the Error stabilizes by 24 samples, and the correlation coefficient with traditional methods exceeds 0.9. Under Low Sampling Conditions, single-symbol computation time is less than 0.01 seconds, making it suitable for real-time processing of Large-scale Symbol Datasets.

- **Consistency with Human Subjective Perception:** On a dataset of 940 symbols with subjective complexity ratings, RLSWSA's perimetric complexity has a Spearman correlation coefficient of 0.74 with human subjective ratings. This shows the algorithm effectively simulates human visual complexity perception, providing a reliable tool for applications like symbol recognition and design optimization.

However, there is still room for improvement. Future work will focus on improving RLSWSA by exploring adaptive sampling strategies to optimize S, researching theoretical bases for the adjustment coefficient α, and evaluating robustness under varying image qualities. To better capture subjective complexity, integrating other perceptual dimensions will also be considered. These advancements aim to enhance RLSWSA's applicability and provide a more flexible and precise tool for font design, visual recognition, UI optimization, and other symbol processing fields.

Acknowledgments

We would like to express our gratitude to all the participants in the experiments and to the reviewers for their valuable feedback.

References

Attneave, F., & Arnoult, M. D. (1956). The quantitative study of shape and pattern perception. *Psychological bulletin*, *53*(6), 452.

Berlyne, D. E., Ogilvie, J. C., & Parham, L. (1968). The dimensionality of visual complexity, interestingness, and pleasingness. *Canadian Journal of Psychology/Revue canadienne de psychologie*, *22*(5), 376.

Chang, L.-Y., Chen, Y.-C., & Perfetti, C. A. (2018). Graphcom: A multidimensional measure of graphic complexity applied to 131 written languages. *Behavior research methods*, *50*, 427–449.

Dai, L., Zhang, K., Zheng, X. S., Martin, R. R., Li, Y., & Yu, J. (2022). Visual complexity of shapes: a hierarchical perceptual learning model. *The Visual Computer*, 1–14.

Donderi, D. C. (2006). Visual complexity: a review. *Psychological bulletin*, *132*(1), 73.

Forsythe, A., Sheehy, N., & Sawey, M. (2003). Measuring icon complexity: An automated analysis. *Behavior Research Methods, Instruments, & Computers*, *35*, 334–342.

Gartus, A., & Leder, H. (2017). Predicting perceived visual complexity of abstract patterns using computational measures: The influence of mirror symmetry on complexity perception. *PloS one*, *12*(11), e0185276.

Guo, X., Qian, Y., Li, L., & Asano, A. (2018). Assessment model for perceived visual complexity of painting images. *Knowledge-Based Systems*, *159*, 110–119.

Majaj, N. J., Pelli, D. G., Kurshan, P., & Palomares, M. (2002). The role of spatial frequency channels in letter identification. *Vision research*, *42*(9), 1165–1184.

Miniukovich, A., & De Angeli, A. (2014). Quantification of interface visual complexity. In *Proceedings of the 2014 international working conference on advanced visual interfaces* (pp. 153–160).

Nag, S., Snowling, M., Quinlan, P., & Hulme, C. (2014). Child and symbol factors in learning to read a visually complex writing system. *Scientific Studies of Reading*, *18*(5), 309–324.

Nagle, F., & Lavie, N. (2020). Predicting human complexity perception of real-world scenes. *Royal Society open science*, *7*(5), 191487.

Pelli, D. G., Burns, C. W., Farell, B., & Moore-Page, D. C. (2006). Feature detection and letter identification. *Vision research*, *46*(28), 4646–4674.

Purchase, H. C., Freeman, E., & Hamer, J. (2012). An exploration of visual complexity. In *Diagrammatic representation and inference: 7th international conference, diagrams 2012, canterbury, uk, july 2-6, 2012. proceedings 7* (pp. 200–213).

Rigau, J., Feixas, M., & Sbert, M. (2005). An information-theoretic framework for image complexity. In *Proceedings of the first eurographics conference on computational aesthetics in graphics, visualization and imaging* (pp. 177–184).

Saraee, E., Jalal, M., & Betke, M. (2020). Visual complexity analysis using deep intermediate-layer features. *Computer Vision and Image Understanding*, *195*, 102949.

Tuch, A. N., Bargas-Avila, J. A., Opwis, K., & Wilhelm, F. H. (2009). Visual complexity of websites: Effects on users' experience, physiology, performance, and memory. *International journal of human-computer studies*, *67*(9), 703–715.

Watson, A. B. (2011). *Perimetric complexity of binary digital images: Notes on calculation and relation to visual complexity* (Tech. Rep.).