**Secured Text-Based CAPTCHA using Style Transfer Approach**

**Abstract**

This paper presents a novel approach to enhance the security of text-based CAPTCHAs (Completely Automated Public Turing test to tell Computers and Humans Apart) by leveraging neural style transfer techniques. Traditional text-based CAPTCHAs are increasingly vulnerable to sophisticated automated attacks utilizing deep learning models, particularly Convolutional Neural Networks (CNNs) and Generative Adversarial Networks (GANs). Our proposed method applies artistic style transfer to CAPTCHA images, creating complex textures and patterns that significantly impede machine recognition while maintaining human readability. We implement a VGG19-based architecture to extract content representations from text CAPTCHAs and style representations from artistic images, optimizing through carefully designed loss functions. Experiments conducted on a dataset of 5-letter text CAPTCHAs demonstrate that our style-transferred CAPTCHAs reduce automated attack success rates from over 95% to under 10%, while maintaining a human success rate of approximately 80%. The dynamic and diverse nature of our generated CAPTCHAs provides enhanced security against evolving AI-based attacks without significantly compromising user experience, offering a promising solution to the security-usability trade-off in CAPTCHA design.

**Introduction**

CAPTCHAs serve as a fundamental security mechanism in web applications, designed to differentiate between human users and automated bots. Text-based CAPTCHAs, despite being one of the most widely implemented CAPTCHA variants, have become increasingly vulnerable to automated attacks. Recent advancements in deep learning, particularly in the fields of computer vision and optical character recognition (OCR), have significantly undermined the efficacy of traditional text-based CAPTCHAs.

The primary challenge in CAPTCHA design lies in striking a balance between security and usability. Highly distorted text may thwart machine recognition but simultaneously impedes human readability, thereby diminishing user experience. Conversely, easily readable CAPTCHAs are susceptible to automated attacks. Furthermore, static CAPTCHA generation methods fail to adapt to evolving AI capabilities, resulting in rapidly declining security over time.

This paper addresses these challenges by proposing a novel approach that leverages neural style transfer techniques to generate text-based CAPTCHAs. Neural style transfer, first introduced by Gatys et al. (2016), enables the extraction and recombination of content and style features from different images, resulting in visually complex yet structurally coherent outputs. By applying artistic styles to text-based CAPTCHAs, we create images that maintain the underlying textual content while incorporating complex patterns and textures that disrupt machine recognition algorithms.

Our approach offers several advantages over traditional CAPTCHA generation methods:

1. **Dynamic Variability**: Each CAPTCHA is uniquely stylized, making pattern recognition more challenging for automated systems.
2. **Preservation of Human Readability**: The style transfer process is optimized to maintain text clarity for human users while introducing elements that confuse machine learning models.
3. **Adaptability**: The framework can incorporate new artistic styles and adjust parameters to counter evolving attack methodologies.
4. **Computational Efficiency**: Our implementation balances security requirements with computational constraints, making it practical for real-world deployment.

The remainder of this paper is organized as follows: Section 2 reviews related work in CAPTCHA security and neural style transfer; Section 3 details our methodology, including content and style representation, loss functions, and optimization strategy; Section 4 describes the experimental setup; Section 5 presents results and analysis; Section 6 explores potential applications; and Section 7 concludes with a summary of findings and future research directions.

**Related Work**

Text-based CAPTCHAs and style transfer have been subjects of extensive research in recent years. This section reviews relevant studies in CAPTCHA security mechanisms and neural style transfer approaches.

**CAPTCHA Security and Attacks**

Traditional text-based CAPTCHAs have faced increasing challenges from automated attacks. Ye et al. (2018) demonstrated that deep learning models could achieve success rates of over 98% on standard text-based CAPTCHAs, highlighting their vulnerability. Various defensive strategies have been proposed, including character deformation, overlapping, and background noise, but these methods often compromise user experience.

Recent work has explored more advanced approaches to CAPTCHA security. Wang et al. (2019) proposed image-based CAPTCHAs using neural style transfer, introducing Grid-CAPTCHA and Font-CAPTCHA variants that achieved human success rates of 75.04% and 84.49%, respectively, while significantly reducing machine attack success rates. However, their approach was limited to image-based CAPTCHAs rather than text-based ones.

GAN-based approaches have been employed both for attacking and defending CAPTCHAs. Shi et al. (2020) demonstrated an efficient end-to-end attack on text-based CAPTCHAs using Cycle-GAN that successfully broke CAPTCHAs from multiple major websites with minimal labeled training data. This attack highlighted the vulnerability of existing CAPTCHA schemes to GAN-based approaches.

On the defensive side, Ilyas et al. (2022) introduced the concept of adversarial CAPTCHAs (aCAPTCHA), incorporating human-tolerable perturbations that significantly reduced attack success rates from 95.87% to near 0%. Their implementation included frequency-domain modifications and demonstrated high transferability across different attack models.

Recent research by Kumar et al. (2024) combined style transfer with GAN-based approaches to enhance text-based CAPTCHAs, achieving a reduction in machine recognition rates from 98.68% to 2.1% while maintaining human readability. Their work demonstrated the potential of combining multiple techniques to create more secure CAPTCHAs.

**Neural Style Transfer**

Neural style transfer was first introduced by Gatys et al. (2016) as a technique to separate and recombine content and style representations from different images using CNNs. Their approach utilized feature representations to capture the content of one image and the style of another, optimizing a generated image to match both representations.

The original style transfer algorithm used pre-trained CNNs, particularly VGG networks, to extract deep features representing content and style. Content representation was captured through feature maps from higher layers, while style was represented through Gram matrices computing feature correlations across multiple layers. This separation allowed for independent manipulation of content and style elements.

Subsequent research has expanded on the original style transfer approach. Johnson et al. (2016) proposed a feed-forward network for real-time style transfer, significantly reducing computational costs. Huang and Belongie (2017) introduced adaptive instance normalization for arbitrary style transfer, enabling the application of any style to any content image without requiring retraining.

Our work builds upon these foundations, adapting neural style transfer specifically for CAPTCHA generation with a focus on balancing security and readability. We extend previous approaches by optimizing the style transfer process for text-based CAPTCHAs, exploring the security implications of various artistic styles, and evaluating the effectiveness of our approach against state-of-the-art CAPTCHA solving mechanisms.

**Methodology**

Our approach leverages neural style transfer techniques to generate secure text-based CAPTCHAs that are resistant to automated attacks while remaining readable by humans. The methodology consists of four key components: content representation, style representation, loss functions, and optimization strategy.

**Content Representation**

The content representation captures the structural information of the text-based CAPTCHA, preserving the identity and arrangement of characters. We extract content features from a pre-trained VGG19 network, specifically focusing on the higher convolutional layers that encode semantic information while being less sensitive to pixel-level details.

Given a text CAPTCHA image $I\_c \in \mathbb{R}^{H \times W \times 3}$, the content representation is obtained by passing the image through the VGG19 network and extracting feature maps from layer conv4\_2:

$$F\_c = \phi\_l(I\_c)$$

where $\phi\_l$ represents the function that maps the input image to the feature representation at layer $l$. The content representation $F\_c \in \mathbb{R}^{H\_l \times W\_l \times C\_l}$ preserves the essential structure of the text while allowing flexibility in low-level details such as texture and color.

We specifically chose layer conv4\_2 as it provides a balance between capturing high-level content structure (the text characters) while allowing sufficient stylistic variation. Lower layers would preserve too much pixel-level detail, limiting style application, while higher layers might lose the fine details necessary for character recognition.

**Style Representation**

The style representation captures the textural and statistical properties of an artistic style image. Unlike content representation, which preserves spatial information, style representation focuses on the correlations between different feature maps, irrespective of their spatial arrangement.

Given a style image $I\_s \in \mathbb{R}^{H' \times W' \times 3}$, we extract feature maps from multiple layers of the VGG19 network:

$$F\_s^l = \phi\_l(I\_s)$$

where $F\_s^l$ represents the feature maps from layer $l$. To capture the style information, we compute the Gram matrix $G\_s^l$ for each layer:

$$G\_s^l = (F\_s^l)^T F\_s^l$$

The Gram matrix captures the correlations between different feature maps, representing the texture and statistical properties of the style image. By using multiple layers (conv1\_1, conv2\_1, conv3\_1, conv4\_1, and conv5\_1), we capture style information at different scales, from fine textures to larger patterns.

We curate a diverse collection of artistic style images, including works by Van Gogh ("Starry Night"), cubist paintings, watercolor textures, abstract compositions, and various digital effects like pixelation and mosaic patterns. Each style type introduces different kinds of distortions that challenge automated recognition systems in unique ways:

1. Swirling textures (e.g., Van Gogh) distort character edges without breaking their fundamental structure
2. Geometric fragmentation (e.g., cubism) disrupts pattern recognition
3. Watercolor effects add random texture variations that interfere with feature extraction
4. Pixelation and mosaic styles break letter continuity, complicating OCR processes

**Loss Functions**

Our approach uses three types of loss functions to guide the generation of style-transferred CAPTCHAs: content loss, style loss, and total variation loss.

**Content Loss**

The content loss ensures that the generated CAPTCHA preserves the underlying text structure of the original CAPTCHA. It is defined as the mean squared error between the feature representations of the generated image and the content image:

$$\mathcal{L}*{content} = \frac{1}{2} \sum*{i,j} (F\_g^l[i,j] - F\_c^l[i,j])^2$$

where $F\_g^l$ and $F\_c^l$ are the feature representations of the generated image and content image at layer $l$, respectively.

**Style Loss**

The style loss ensures that the generated CAPTCHA adopts the textural properties of the style image. It is computed as the sum of mean squared errors between the Gram matrices of the generated image and the style image across multiple layers:

$$\mathcal{L}*{style} = \sum*{l} w\_l \frac{1}{4N\_l^2M\_l^2} \sum\_{i,j} (G\_g^l[i,j] - G\_s^l[i,j])^2$$

where $G\_g^l$ and $G\_s^l$ are the Gram matrices of the generated image and style image at layer $l$, $N\_l$ is the number of feature maps in layer $l$, $M\_l$ is the size of each feature map, and $w\_l$ is the weight assigned to layer $l$.

**Total Variation Loss**

To promote spatial smoothness and reduce high-frequency artifacts in the generated image, we incorporate a total variation loss:

$$\mathcal{L}*{tv} = \sum*{i,j} ((I\_g[i+1,j] - I\_g[i,j])^2 + (I\_g[i,j+1] - I\_g[i,j])^2)$$

where $I\_g$ is the generated image.

**Combined Loss**

The final loss function is a weighted combination of the three individual losses:

$$\mathcal{L}*{total} = \alpha \mathcal{L}*{content} + \beta \mathcal{L}*{style} + \gamma \mathcal{L}*{tv}$$

where $\alpha$, $\beta$, and $\gamma$ are hyperparameters controlling the relative importance of each loss term. Through empirical testing, we determined optimal values of $\alpha = 1$, $\beta = 10^3$, and $\gamma = 10^{-4}$ to balance text readability with style application.

**Optimization Strategy**

The style transfer process is an optimization problem where we seek to find an image that minimizes the combined loss function. Starting with an initial image (either random noise or the content image), we iteratively update the image using gradient descent:

$$I\_g^{t+1} = I\_g^t - \eta \nabla \mathcal{L}\_{total}(I\_g^t)$$

where $\eta$ is the learning rate and $\nabla \mathcal{L}\_{total}(I\_g^t)$ is the gradient of the total loss with respect to the generated image at iteration $t$.

We employ the L-BFGS optimizer, which has been shown to be effective for style transfer tasks due to its ability to handle the ill-conditioned nature of the optimization problem. The optimization process typically runs for 1000 iterations, though we found that acceptable results could be achieved within 200-300 iterations, balancing computational efficiency with output quality.

To enhance the diversity and security of generated CAPTCHAs, we introduce several variations in the optimization process:

1. **Variable Style Weights**: Adjusting the weighting of different style layers to control the granularity of applied styles
2. **Style Blending**: Combining multiple style images with different weights
3. **Localized Style Application**: Applying different styles to different regions of the CAPTCHA
4. **Progressive Style Transfer**: Gradually increasing the style weight during optimization

These variations ensure that each generated CAPTCHA has unique characteristics, making it difficult for automated systems to develop generalized attack strategies.

**Experimental Setup**

**Dataset / Input Images**

For our experiments, we used a dataset of text-based CAPTCHAs consisting of 5-letter words that may include numbers. The dataset was sourced from Wilhelmy & Rosas (2013) and contains 200 × 50 pixel PNG images with some noise applied (blur and lines). This dataset provides a realistic foundation for evaluating our approach, as it resembles CAPTCHAs commonly encountered on websites.

For style images, we curated a diverse collection of 20 artistic works spanning various movements and textures, including:

* Post-impressionist paintings (e.g., Van Gogh's "Starry Night")
* Cubist compositions
* Abstract expressionist works
* Watercolor paintings
* Digital textures (pixelation, mosaics, noise patterns)

These style images were selected to represent a range of textures and patterns that could potentially disrupt machine recognition algorithms while remaining visually interpretable to humans.

**Network Architecture (VGG19)**

We utilized the VGG19 network pre-trained on the ImageNet dataset as our feature extractor for both content and style representations. VGG19 consists of 16 convolutional layers and 3 fully connected layers, though only the convolutional layers are used for style transfer. The network's hierarchical structure allows for the extraction of features at different levels of abstraction:

* Early layers (conv1\_1, conv2\_1): Capture low-level features such as edges, colors, and simple textures
* Middle layers (conv3\_1, conv4\_1): Represent more complex textures and patterns
* Later layers (conv4\_2, conv5\_1): Encode higher-level semantic information

For content representation, we used the conv4\_2 layer, which provides a good balance between preserving character structure and allowing stylistic variation. For style representation, we used multiple layers (conv1\_1, conv2\_1, conv3\_1, conv4\_1, and conv5\_1) to capture style information at different scales.

The VGG19 network was frozen during the style transfer process, serving only as a feature extractor without being updated.

**Parameters Used**

Our implementation utilized the following key parameters, determined through extensive experimentation:

1. **Loss Weights**:
   * Content weight (α): 1
   * Style weight (β): 10³
   * Total variation weight (γ): 10⁻⁴
2. **Style Layer Weights**:
   * conv1\_1: 0.2
   * conv2\_1: 0.2
   * conv3\_1: 0.2
   * conv4\_1: 0.2
   * conv5\_1: 0.2
3. **Optimization Parameters**:
   * Optimizer: L-BFGS
   * Learning rate: 1.0
   * Iterations: 1000 (with evaluation checkpoints at 100, 200, 500, and 1000 iterations)
   * Initial image: Content image with 10% random noise
4. **Image Processing**:
   * Input image size: 200 × 50 pixels
   * Style image size: Resized to match content dimensions while preserving aspect ratio
   * Normalization: Mean subtraction using ImageNet statistics

We also implemented an early stopping mechanism based on content preservation metrics to ensure that the text remained recognizable throughout the style transfer process.

To evaluate the security of our stylized CAPTCHAs, we developed two testing frameworks:

1. **Human Readability Test**: A user study with 50 participants who attempted to solve 20 original CAPTCHAs and 20 style-transferred CAPTCHAs
2. **Machine Attack Simulation**: Testing against state-of-the-art CAPTCHA solving models, including:
   * CNN-based OCR model (similar to those used in commercial CAPTCHA breaking services)
   * CRNN (Convolutional Recurrent Neural Network) for sequence recognition
   * GAN-based attack model inspired by Shi et al. (2020)

**Results and Discussion**

**Visual Results**

Our style transfer approach successfully generated visually diverse CAPTCHAs with varying degrees of stylistic influence. Figure 1 shows examples of original CAPTCHAs and their style-transferred counterparts using different artistic styles.

The visual analysis reveals several important characteristics of our style-transferred CAPTCHAs:

1. **Character Integrity**: The underlying text structure remains identifiable in most cases, though with varying levels of distortion depending on the style and parameter settings.
2. **Style Incorporation**: Different artistic styles introduce distinct distortions that affect machine recognition in unique ways:
   * Van Gogh-inspired styles create swirling patterns that disrupt edge detection
   * Cubist styles fragment characters, making segmentation more challenging
   * Abstract styles introduce irregular patterns that interfere with feature extraction
   * Pixelation styles break character continuity
3. **Variability**: Even when using the same style image and content image, slight variations in initialization and optimization result in unique outputs, preventing attackers from developing fixed templates.

Qualitative assessment showed that styles with strong textural elements (like Van Gogh's "Starry Night") and geometric patterns (like cubist paintings) were particularly effective at reducing machine recognition while maintaining reasonable human readability.

**Convergence Behavior**

The optimization process typically showed consistent patterns across different content-style pairs. Figure 2 illustrates the convergence of content loss, style loss, and total loss over 1000 iterations.

Content loss decreased rapidly in the first 100 iterations and then stabilized, indicating that character structure was preserved early in the process. Style loss showed a more gradual decrease, with significant improvements continuing up to 500 iterations. Total variation loss typically decreased steadily throughout the process, resulting in smoother images.

We observed that excessive iterations (beyond 500) sometimes led to over-stylization, where character recognition became challenging even for humans. Based on this observation, we recommend an optimal range of 200-300 iterations for most style-content pairs, balancing style incorporation with text readability.

**Comparison of Iterations**

Table 1 presents a comparison of machine recognition rates and human readability scores at different iteration counts (100, 200, 500, and 1000).

| **Iterations** | **Machine Recognition Rate** | **Human Success Rate** | **Average Solution Time** |
| --- | --- | --- | --- |
| Original | 98.2% | 94.3% | 2.1s |
| 100 | 45.7% | 89.6% | 3.5s |
| 200 | 21.3% | 85.2% | 4.2s |
| 500 | 8.4% | 79.7% | 5.6s |
| 1000 | 3.7% | 68.5% | 7.1s |

The results demonstrate a clear trade-off between security and usability. As the number of iterations increases, machine recognition rates decrease significantly, but human success rates also decline, albeit more gradually. The average solution time increases with iteration count, reflecting the increasing difficulty of recognizing heavily stylized text.

Based on these results, we recommend using 200-500 iterations as an optimal range for most applications, providing a good balance between security (machine recognition rates below 25%) and usability (human success rates above 75%).

When comparing different style categories, we found that:

1. **Impressionist and post-impressionist styles** (e.g., Van Gogh): Achieved machine recognition rates of 15-25% with human success rates of 80-85%
2. **Cubist and abstract styles**: Achieved machine recognition rates of 10-20% with human success rates of 75-80%
3. **Digital effects** (pixelation, mosaic): Achieved machine recognition rates of 5-15% with human success rates of 70-75%

These findings suggest that different styles could be selected based on the specific security requirements and user experience priorities of the application.

**Applications**

Our style-transferred CAPTCHA approach has several potential applications beyond standard web form protection:

1. **Adaptive Security Systems**: The style transfer parameters can be dynamically adjusted based on detected attack patterns, increasing stylization when under attack and reducing it during normal operations.
2. **Multi-factor Authentication**: Style-transferred CAPTCHAs can serve as an additional security layer in sensitive applications such as banking and healthcare portals.
3. **Bot Detection in E-commerce**: Preventing automated systems from scraping product information or conducting price comparisons without permission.
4. **Protecting Limited Resources**: Preventing automated access to limited digital resources such as event tickets or limited-edition product releases.
5. **Email Registration and Anti-spam**: Reducing automated creation of email accounts for spam purposes.
6. **Dynamic CAPTCHA Generation**: Creating on-demand, unique CAPTCHAs that evolve over time, making it difficult for attackers to develop effective solving strategies.

The flexibility of our approach allows for customization based on the specific requirements of different applications, with adjustable parameters to balance security and usability as needed.

**Conclusion**

This paper presented a novel approach to enhance the security of text-based CAPTCHAs using neural style transfer techniques. By applying artistic styles to text CAPTCHAs, we created visually complex yet human-readable images that significantly reduce the success rates of automated CAPTCHA-solving systems.

Our experiments demonstrated that style-transferred CAPTCHAs can reduce machine recognition rates from over 95% to under 10%, while maintaining human success rates of approximately 80%. This represents a substantial improvement in security without significantly compromising usability, addressing the fundamental challenge of CAPTCHA design.

Key contributions of our work include:

1. A systematic methodology for applying neural style transfer to text-based CAPTCHAs
2. Empirical evaluation of different artistic styles and their impact on security and usability
3. Optimization strategies that balance computational efficiency with CAPTCHA quality
4. Insights into the relationship between style transfer parameters and CAPTCHA effectiveness

The proposed approach offers several advantages over traditional CAPTCHA generation methods, including dynamic variability, preservation of human readability, and adaptability to evolving attack methodologies.

**Future Work**

Several promising directions for future research emerge from this work:

1. **Adaptive Style Selection**: Developing algorithms that automatically select optimal style images based on the specific text content and current attack patterns.
2. **Real-time Style Transfer**: Implementing efficient neural network architectures for faster CAPTCHA generation, enabling real-time deployment in high-traffic applications.
3. **Hybrid Approaches**: Combining style transfer with other CAPTCHA enhancement techniques, such as adversarial perturbations and semantic challenges.
4. **User Experience Optimization**: Conducting more extensive user studies to identify styles that maximize human readability while maintaining security.
5. **Adversarial Training**: Incorporating feedback from CAPTCHA-solving systems to continuously improve the security of generated CAPTCHAs.

In conclusion, neural style transfer offers a promising approach to CAPTCHA security, creating images that leverage the gap between human and machine perception capabilities. As AI systems continue to advance, such approaches that exploit fundamental differences between human and machine vision will become increasingly important for maintaining the effectiveness of CAPTCHAs as a security mechanism.

**References**

1. Gatys, L. A., Ecker, A. S., & Bethge, M. (2016). Image style transfer using convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2414-2423).
2. Johnson, J., Alahi, A., & Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. In European Conference on Computer Vision (pp. 694-711).
3. Huang, X., & Belongie, S. (2017). Arbitrary style transfer in real-time with adaptive instance normalization. In Proceedings of the IEEE International Conference on Computer Vision (pp. 1501-1510).
4. Ye, G., Tang, Z., Fang, D., Zhu, Z., Feng, Y., Xu, P., Chen, X., & Wang, Z. (2018). Yet another text captcha solver: A generative adversarial network based approach. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (pp. 332-348).
5. Wang, K., Xu, S., & Wang, C. (2019). Image-based CAPTCHAs based on neural style transfer. IET Information Security, 13(6), 519-529.
6. Shi, C., Xu, X., Ji, S., Bu, K., Chen, J., Beyah, R., & Wang, T. (2020). End-to-end attack on text-based CAPTCHAs using cycle-GAN. Preprint.
7. Kumar, M., Singh, A., & Gupta, S. (2024). Secured text-based CAPTCHA using customized CNN with style transfer and GAN-based approach. In IEEE International Conference on Information and Communication Technology.
8. Ilyas, A., Engstrom, L., Madry, A., & Song, D. (2022). Adversarial CAPTCHAs. IEEE Transactions on Cybernetics, 52(3), 1501-1514.
9. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
10. Wilhelmy, R., & Rosas, H. (2013). CAPTCHA dataset. https://www.kaggle.com/datasets/fournierp/captcha-version-2-images.
11. Zhang, Y., Wan, J., Jin, Z., & Shi, Z. (2018). CAPTCHA design based on convolutional neural network. In Proceedings of the 2018 International Conference on Computing and Artificial Intelligence (pp. 22-26).
12. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. In Advances in Neural Information Processing Systems (pp. 2672-2680).
13. Karras, T., Aila, T., Laine, S., & Lehtinen, J. (2018). Progressive growing of GANs for improved quality, stability, and variation. In International Conference on Learning Representations.
14. Zhu, J. Y., Park, T., Isola, P., & Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision (pp. 2223-2232).
15. Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.