engineering proceedings

MDPI

Proceeding Paper

A Linear Differentiation Scheme for Camouflaged Target Detection using Convolution Neural Networks †

Jagadesh Sambbantham [1,*], Gomathy Balasubramanian [2], Rajarathnam [3] and Mohit Tiwari [4]

[1] Department of ECE, KPR Institute of Engineering and Technology, Coimbatore 641407, Tamilnadu, India
[2] Department of CSE, Dr.N.G.P Institute of Technology, Coimbatore 641048, Tamilnadu, India; gomramesh@gmail.com
[3] Department of Mechatronics Engineering, Paavai Engineering College, Namakkal 637018, Tamilnadu, India; drprajamala@gmail.com
[4] Department of CSE, Bharati Vidyapeeth's College of Engineering, Delhi 110063, India; mohit.tiwari@bharatividyapeeth.edu
* Correspondence: jackshree@gmail.com
† Presented at the International Conference on Recent Advances in Science and Engineering, Dubai, United Arab Emirates, 4–5 October 2023.

Abstract: Camouflaged objects are masked within an existing image or video under similar patterns. This makes it tedious to detect target objects post classification. The pattern distributions are monotonous due to similar pixels and non-contrast regions. In this paper, a distribution-differentiated target detection scheme (DDTDS) is proposed for segregating and identifying camouflaged objects. First, the image is segmented using textural pixel patterns for which the linear differentiation is performed. Convolutional neural learning is used for training the regions across pixel distribution and pattern formations. The neural network employs two layers for linear training and pattern differentiation. The differentiated region is trained for its positive rate in identifying the region around the target. Non-uniform patterns are used for training the second layer of the neural network. The proposed scheme pursues a recurrent iteration until the maximum segmentation is achieved. The metrics of positive rate, detection time, and false negatives are used for assessing the proposed scheme's performance.

Keywords: camouflaged target detection; convolution neural network; linear differentiation; pattern detection; segmentation

check for updates

Citation: Sambbantham, J.; Balasubramanian, G.; Rajarathnam; Tiwari, M. A Linear Differentiation Scheme for Camouflaged Target Detection using Convolution Neural Networks. *Eng. Proc.* **2023**, *59*, 45. https://doi.org/10.3390/engproc2023059045

Academic Editors: Nithesh Naik, Rajiv Selvam, Pavan Hiremath, Suhas Kowshik CS and Ritesh Ramakrishna Bhat

Published: 13 December 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Camouflaged object detection (COD) is a process that detects the objects that are presented in videos and images. Important patterns and parameters are detected from the images using COD. A rethinking COD method is commonly used to detect the objects from the images [1]. The COD method detects the visual effects, patterns, color, texture, and motion features of the images. The COD method uses a feature-based structure to analyze the depth and content of the objects [2]. The COD method also identifies the disguised targets that improve the functional performance range of the systems. The neighbor connection-based COD method is mainly used to detect the hidden similarities in the background in the images [3]. Hierarchical information transfer (HIT) is used in a method that identifies the neighboring layers in the images. The HIT-based COD method enhances the effectiveness level of the object detection process [4]. A semi-supervised search identification network-based COD approach is also used in the detection process. Both spatial and temporal features are detected in the images. The textual and statistical parameters are also detected, which produce optimal information for the COD process [5].

Machine learning (ML) algorithms and techniques are used for the COD process. ML is mainly used to increase the accuracy level of the object detection process. A polarization

imaging and deep learning (DL)-based COD method is used for the applications. A self-supervised DL algorithm is used here to detect the camouflaged targets from the given outputs. The polarization imaging detects the important parameters using a segmentation algorithm. The DL algorithm improves the effectiveness and performance range in target detection processes [6,7]. A deep gradient learning network (DGNet) is used for COD. The DGNet analyzes the context and texture features from the images. The DGNet detects the transparent and hidden objects which are presented in the images. The DGNet provides effective datasets for detection that reduce the latency in the computation process. The COD method also identifies the similarities among the objects that reduce the computational cost ratio of the systems [8]. The contributions of this study are summarized as follows:

- Designing a target object detection scheme in camouflaged images using segmentation and variation detection with high positive rates;
- Incorporating the convolution neural network process with modified layer operations for differentiation detection and training;
- Performing experimental and comparative analysis using external inputs and metrics for assessing the proposed scheme's performance.

2. Related Works

Yuanying et al. [9] introduced a camouflage detection approach using local minimum difference constraints (LMDC). The mirror differences are identified based on the moving target from the given input data. The introduced method removes the stable background region from the targets. It also calculates the attributes based on the details of the pixels, which reduces the latency in the computation process.

Zhai et al. [10] proposed a new mutual graph learning (MGL) model for camouflaged object detection. Task-specific feature maps are used in the model which are necessary for detection. The MGL model detects the actual relationship between the interaction process. It provides higher-resolution services to the users that minimize the energy consumption level in object detection.

Xu et al. [11] designed a new boundary guidance network for camouflage object detection (COD). A localization decoder is implemented in the method to capture multi-scale information for further processes. The localization decoder also identifies the missing parts of objects. A residual refinement is used here to enhance the region branch of the objects. The designed network increases the accuracy range in COD, which improves the effectiveness ratio of the systems.

Liu et al. [12] introduced a bi-level recurrent refinement network (Bi-RRNet) for COD systems. The main aim of the network is to analyze the global information which is presented in the content. Both spatial and temporal features are detected that produce optimal information for the detection process. High-level semantic features are evaluated using an enhancement module that reduces the complexity of the identification process. The introduced network maximizes the accuracy in object detection, which enhances the performance level of the applications.

Ji et al. [13] developed an edge-based reversible re-calibration network (ERRNet)-based COD method. A selection edge aggregation (SEA) module is implemented in the network to fetch prior edges of the nodes and pixels. The visual behaviors of the objects and scenes are also detected in the images.

Bi et al. [14] proposed a COD framework named CFNet, CFNet, which is mainly used to solve problems in the detection process. The actual goal of the framework is to reduce the blurred edges and difficulties in object detection. The contradiction area detection module (CADM) and feature aggregation module (FAM) are used to address the important data that are relevant to the objects. The proposed framework improves the feasibility and effectiveness level of COD systems.

Zhang et al. [15] designed a COD method using a transformer-included progressive refinement network (TPRNet). TPRNet detects the semantic and spatial parameters and information for the detection process. A refinement concurrency unit (RCU) is obtained

here to prior the guidance for object detection. The RCU module provides rich parameters to detect the details of the objects for further processes. The designed framework increases the performance range of the systems.

Xiao et al. [16] proposed a boundary-guided context-aware network (BCNet) for COD. The proposed network is a high-resolution object detection method that minimizes the complexity of the detection process. The BCNet identifies the multi-scale information and details of objects that are presented in the scenes. It boosts the inner layer of the detection process, which improves the reliability range of the systems. The proposed BCNet increases the real-time speed capacity in COD systems.

Apart from the above discussion, the problem of camouflaged boundary detection and the variation estimation problems are considered in this article. The boundary-based approach in [16] requires certain textural classifications regardless of the image size and pixel representations. The contradiction method discussed in [14] requires deviating instances that segregate the pixels, resulting in false negatives. Such issues are addressed in this proposed scheme for improving the positive rate of object detection.

3. Distribution-Differentiated Target Detection Scheme

This proposed distribution-differentiated target detection scheme (DDTDS) helps in conveying the challenge of determining camouflaged objects within images or videos, where the target objects are masked by similar patterns and distributions, making it crucial to identify them precisely after classification. By manipulating textural pixel patterns, convolutional neural learning, and an iterative training process, this process efficaciously differentiates and determines camouflaged objects with enhanced positive rates and reduced false negatives. Furthermore, the use of non-uniform patterns in the training procedure helps in enhancing the process's ability to detect diverse and complex camouflaged objects. A schematic illustration of the proposal is presented in Figure 1.

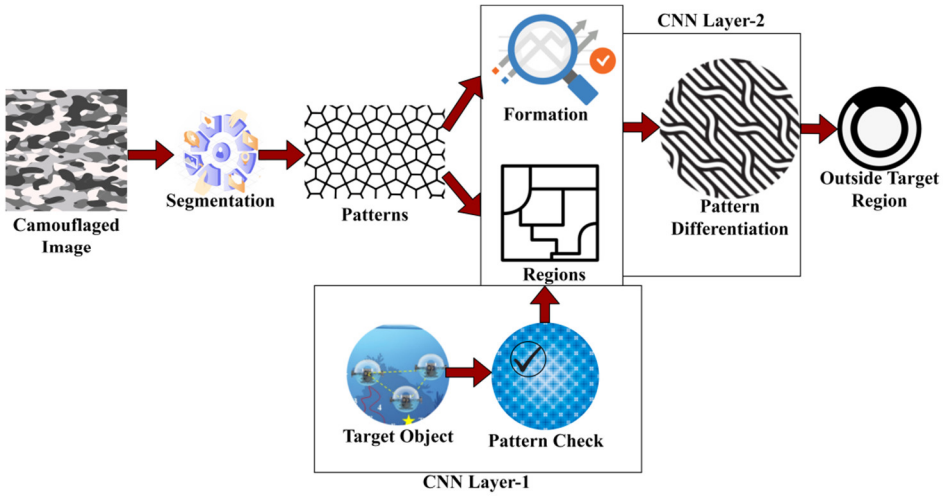


Figure 1. Schematic illustration of the DDTD scheme.

The camouflaged images are analyzed for the further segmentation procedure, and to extract patterns from the segmentation. This proposed DDTDS helps in determining the region that contains the potential camouflaged objects. This input analysis process helps in scrutinizing their linear differentiation to extract the valuable characteristics of the images and their distinctions between the target images and then the background. After determining the camouflage images, the segmentation process is performed according to it. The segmentation operation is employed to separate the camouflaged object from the surrounding background. This involves using the information which is obtained from the linear differentiation of the textural pixel patterns to describe the bounds of the potential

target regions. The process of determining the camouflage image via the segmentation process is explained by the following equation given below:

$$\left.\begin{array}{c} X = \overline{x} \\ \overline{x} = \left(\overline{x}_{ij} + i\overline{y}_{ij}\right) \\ = \left(\left(\underline{x}_{ij}(a), \overline{x}_{ij}(a)\right) + i\left(\underline{y}_{ij}(a), \overline{y}_{ij}(a)\right)\right) \\ where\ \ 1 \le \underline{i},\ \ j \le n,\ \ 0 \le a \le 1, \\ \overline{A} = \left(\overline{a}_{ij}\right) \\ \overline{A} = \left(\underline{a}_{ij}\right) \\ = \left(\underline{a}_{ij}(a), \overline{a}_{ij}(a)\right) + i\left(\underline{b}_{ij}(a), \overline{b}_{ij}(a)\right) \end{array}\right\} \quad (1)$$

where x denotes the input image, y represents the textural patterns, A denotes the segmentation operation in detecting the camouflaged object, i represents the characteristics of an image, and j denotes the target regions. After the segmentation process, the pattern determination procedure is performed to extract the formation and regions of the target input images. This process involves training the neural network to distinguish between uniform and non-uniform patterns. Uniform textural patterns represent the background regions, while non-uniform patterns denote the presence of the potential camouflage object in the target images. The convolutional neural network helps in analyzing the pattern formations and distribution of the pixels within each segmented region. It extracts the patterns for every segmented region. The process of determining the patterns of segmented regions is explained by the following equation given below:

$$\left.\begin{array}{c} \overline{X} = \overline{M} + i\overline{N} \\ \overline{Y} = \overline{a} + i\overline{b} \\ \overline{X} + \overline{Y} = \left(\overline{a} + \overline{b}\right) + i\left(\overline{M} + \overline{N}\right) \\ \overline{X} - \overline{Y} = \left(\overline{a} - \overline{b}\right) + i\left(\overline{M} - \overline{N}\right) \\ a\overline{X} = a\overline{Y} + ia\overline{M}, \quad M \in N \\ \overline{X} \times \overline{Y} = \left(\overline{M} \times \overline{N} - \overline{a} \times \overline{b}\right) + i\left(\overline{a} \times \overline{b} + \overline{M} \times \overline{N}\right) \\ X = \left[\underline{M}(a), \overline{M}(a)\right] + i\left[\underline{N}(a), \overline{N}(a)\right] \\ = \left(\left[a_{ij}(a), \overline{a}_{ij}(a)\right]\right. \\ where\ \ 0 \le a \le 1 \end{array}\right\} \quad (2)$$

where M represents the patterns of the segmented regions, and N denotes the training operation. Now, the formation and regions of the camouflaged images are determined for further processes. This process ensures that the determined images are formed precisely according to the input images. They are identified for further training processes and also to determine the outside target regions. This ensures that the identified formation accurately corresponds to the potential camouflaged objects. The process of determining the formation of the images is explained by the following equation given below:

$$\left.\begin{array}{c} \begin{cases} KX(a)B = C(a) \\ KY(a)B = D(a) \end{cases} \\ K = \begin{pmatrix} X^{+} & -X^{-} \\ -X^{-} & X^{+} \end{pmatrix} \\ X(a) = \begin{pmatrix} \underline{M}(a) \\ -\overline{M}(a) \end{pmatrix} \\ B(a) = \begin{pmatrix} \underline{X}(a) \\ -\overline{X}(a) \end{pmatrix} \\ Y(a) = \begin{pmatrix} \underline{N}(a) \\ -\overline{N}(a) \end{pmatrix} \\ C(a) = \begin{pmatrix} \underline{Z}(a) \\ -\overline{Z}(a) \end{pmatrix} \end{array}\right\} \quad (3)$$

where K represents the formation of the image, and C represents the precise information of its origination. The image formation using the patterns is illustrated in Figure 2.

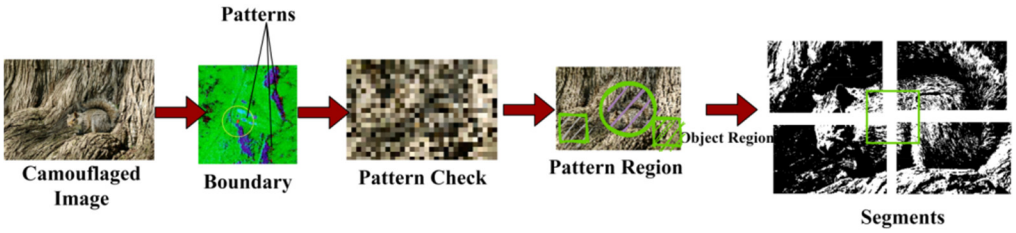


Figure 2. Image formation using patterns.

The input camouflaged image is differentiated using y for multiple pixel regions. As pixels meet $x, y = max$, the i is used for pattern checking across multiple regions. The CNN's first layer is responsible for verifying the obtained patterns across various j and M regions through N. If the N is non-deviating (incremental), then K is obtained from $i \in C$. The segments are thus classified from the patterns obtained; the patterns are used for image formation (Figure 2). Now, the regions of the images are determined in this process. They are identified for the further verification operation of the targeted objects. These are the procedures conducted to reduce false positives and to improve the detection accuracy of the process. They help in achieving image recognition depending on the training of the network, and thus this process is explained by the following equation given below:

$$\left.\begin{array}{c} \begin{cases} A(n) = \frac{1}{2}(A^+ - A^-) + (A^+ + A^-) \\ B(n) = \frac{1}{2}(B^+ - B^-) + (B^+ + B^-) \end{cases} \\ S = \begin{pmatrix} \underline{x}(a) & \underline{y}(a) \\ \overline{x}(a) & \overline{y}(a) \end{pmatrix} B \\ X = \begin{pmatrix} \underline{M}(a) - \overline{N}(a) \\ \underline{N}(a) - \underline{M}(a) \end{pmatrix} A \end{array}\right\} \tag{4}$$

where S represents the region of the camouflage images. Now, the CNN first layer helps in checking the pattern of the target object. The training process is performed for the neural network to extract the formation and region of the target objects. Then, it is compared with the camouflaged image's formation and region. After the training process, the neural network helps in extracting the distinctive features and formations of the region surrounding the target. In this CNN first-layer process, a procedure is performed to check whether the camouflaged formation and region are the same as the formation and region of the camouflaged image. The differentiated region is trained for its positive rate in identifying the region around the target. The process of the first layer of CNN is explained by the following equation given below:

$$\left.\begin{array}{c} SS^+ = \begin{pmatrix} A^+ & -A^- \\ -A^- & A^+ \end{pmatrix} \begin{pmatrix} a & b \\ b & a \end{pmatrix} = \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix} \\ A^+a - A^-b = I \\ A^+a - A^-b = 0 \\ a + b = [A^+ - A^-] \\ a - b = [A^+ + A^-] \\ \begin{pmatrix} A^+ & -A^- \\ -A^- & A^+ \end{pmatrix} \begin{pmatrix} \underline{P}(x) & \underline{Q}(x) \\ \overline{P}(x) & \overline{\overline{Q}}(x) \end{pmatrix} B = \begin{pmatrix} \underline{M}(a) & \underline{N}(a) \\ \overline{M}(a) & \overline{N}(a) \end{pmatrix} \\ \begin{cases} AX(a)P = P(a) \\ AY(a)Q = Q(a) \end{cases} \\ P = \begin{pmatrix} B^+ & -B^- \\ -B^- & B^+ \end{pmatrix} \end{array}\right\} \tag{5}$$

where P represents the operation of the CNN layer 1 in this process. Now the pattern differentiation is determined in the second layer of the CNN. The non-uniform patterns are used for training the second layer of the neural network. These non-uniform patterns denote the difficult and precise characteristics present in the camouflaged images. By providing efficient training to the neural network with such patterns, it identifies the variations in the target object. After determining the pattern differentiation, the outside target region is extracted. Hence, the CNN second-layer process is explained by the following equation given below:

$$\left.\begin{array}{c} \underline{V}(a) = A(\overline{X})a - A(\underline{X})a \\ \overline{V}(a) = -A(\overline{Y})b - A(\underline{Y})b \\ B = \frac{1}{2}((B^+ - B^-) + (B^+ + B^-)) \\ B(A) == \frac{1}{2}((B^+ - B^-) - (B^+ + B^-)) \\ \begin{cases} SX(a)B = C(a) \\ SY(a)B = D(a) \end{cases} \\ S = \begin{pmatrix} A^+ & -A^- \\ -A^- & A^+ \end{pmatrix} \\ A(x) = \begin{pmatrix} \underline{X}(a) \\ -\overline{X}(a) \end{pmatrix} \\ B(y) = \begin{pmatrix} \underline{Y}(a) \\ -\overline{Y}(a) \end{pmatrix} \end{array}\right\} \quad (6)$$

where V represents the pattern differentiation in the second layer of CNN. Now, the outside target region is determined after the two-layer process of the CNN. With both layers' outcomes, this CNN helps in precisely delineating the outside of the target region. The second-layer process of the CNN is illustrated in Figure 3.

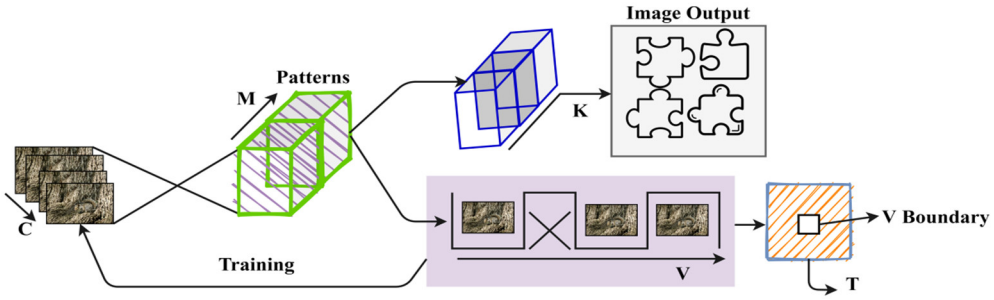


Figure 3. Second-layer process of the CNN.

The CNN's second layer is responsible for differentiating V from the input image. The mixed M and C are used for concurrent layer-based analysis through N in the first layer. In this process, either $f(x)$ or $B(y)$ is used for validating V throughout C. If $i \notin C$ is observed anywhere, the variation boundaries are identified using the pixels present (Refer to Figure 3). The outside regions of the targeted images are consolidated as the background region. This procedure helps in ensuring the CNN [17–19] aims at the target area, establishing precise and efficacious target detection within the camouflaged images. The process of extracting the outside target region is explained by the following equation given below:

$$\left.\begin{array}{c} T = (X(Y) * A(B)) \\ = \frac{1}{2}\begin{pmatrix} (B^+ + B^-) + (B^+ - B^-) & (B^+ + B^-) - (B^+ - B^-) \\ (B^+ + B^-) - (B^+ - B^-) & (B^+ + B^-) + (B^+ - B^-) \end{pmatrix} \\ \begin{pmatrix} \underline{M}(a) & -\overline{M}(a) \\ \underline{N}(a) & -\overline{N}(a) \end{pmatrix} = A\begin{pmatrix} \underline{x}(a) & -\overline{x}(a) \\ \underline{y}(a) & -\overline{y}(a) \end{pmatrix}\begin{pmatrix} B^+ & -B^- \\ -B^- & B^+ \end{pmatrix} \end{array}\right\} \quad (7)$$

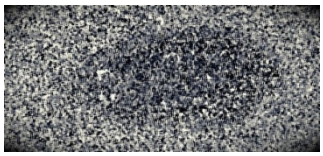
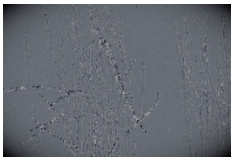
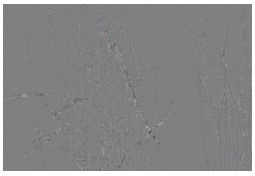
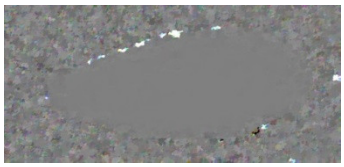
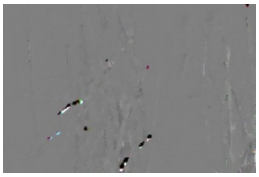
where T represents the outside target region. This proposed system helps in detecting the outside target region by assessing the positive rate, detection time, and false negatives. This helps to reduce unnecessary false information and the time taken for the detection of the

outside target region. The patterns of the camouflaged image are detected and compared with the targeted objects to extract the pattern differentiation. Additionally, based on this output, the outside target region is determined.

4. Experimental Analysis

In this sub-section, the images from [20] are used for validating the functions of the proposed scheme. This reference provides 455 camouflage images for testing and training. which are related to underwater life. From the given input set, we consider a few images for describing the proposed scheme's process, as presented in Table 1.

Table 1. Experimental output illustration.

Input		
Segments		
Patterns		
Object Detection		
Outside Region		

5. Results and Discussion

The results and discussion in this section are presented using the positive rate, detection time, and false negatives observed in the experimental analysis. The number of patterns and regions varied up to 24 and 9, respectively in this analysis. The methods BGN-COD [11], and ERR-Net [13] are used along the proposed scheme in this comparative analysis.

5.1. Positive Rate

The metric positive rate defines the maximum differentiation score between y and S for T detection. This is computed as $positiverate = \frac{y+s}{A+M} \forall i \in C$. The positive rate is efficacious in this method, with the precise camouflaged input image and then its textural patterns. The segmentation process is performed according to the input image, and thus it helps in the further formation and region determination operation. This process efficaciously differentiates and determines camouflaged objects with enhanced positive rates and reduced false negatives. By using CNN, the pattern differentiation is identified for the extraction of the outside target region (Figure 4).

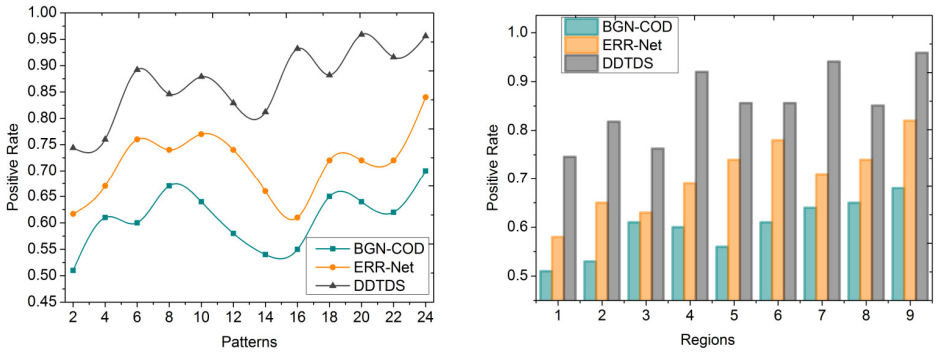


Figure 4. Positive rate analysis.

5.2. Detection Time

The time taken for the detection is lesser in this process, with the help of the two layers of the CNN. The pattern determination procedure is performed to extract the formation and regions of the target input images. This process involves training the neural network to distinguish between uniform and non-uniform patterns. The differentiated region is trained for its positive rate in identifying the region around the target. By consolidating the mentioned characteristics, the detection time is reduced (Figure 5).

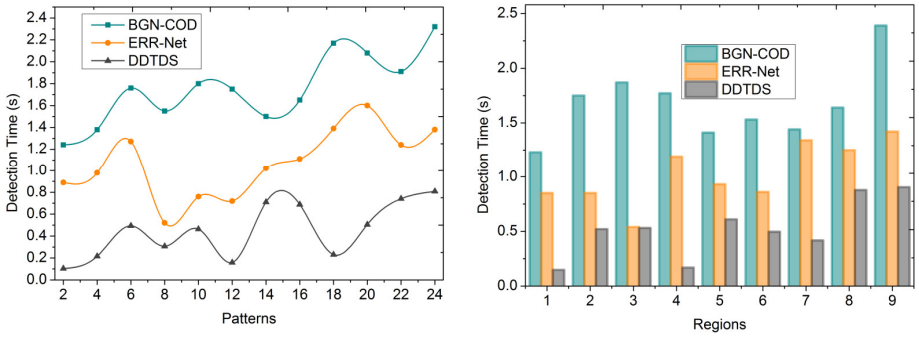


Figure 5. Detection time analysis.

5.3. False Negatives

The false negatives are those V that fail to identify the target regardless of $y \in M$. This means the patterns are not identified during the intersection, and hence it is computed as $\frac{M-y}{M} \forall\, i \notin C$. The false negatives are lesser in this process due to the enhanced precision of the obtained patterns. By manipulating textural pixel patterns, convolutional neural learning and its training process efficaciously differentiate and determine camouflaged objects with enhanced positive rates and reduced false negatives. These are procedures conducted to reduce false positives and to improve the detection accuracy of the process (Figure 6).

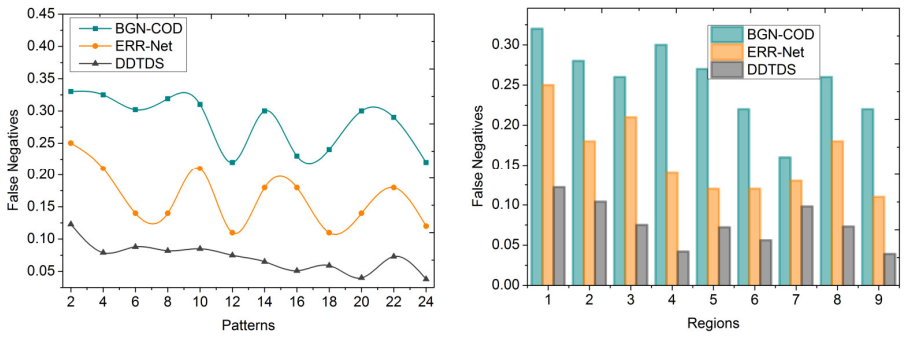


Figure 6. False negatives' analysis.

6. Conclusions

The DDTDS commences by separating the input image using textural pixel patterns, and then it performs linear differentiation on these patterns. This process helps to differentiate the regions that contain prospective camouflaged objects from the background. To attain this, a CNN is engaged in training the regions based on pixel distribution and pattern formations. The neural network consists of two layers, and its primary aim is to distinguish the patterns linearly. In the first layer, the differentiated region is trained to enhance its positive rate in determining the area around the target. The second layer of the neural network is trained using non-uniform patterns. Therefore, this proposed scheme engages are peated iteration process to continuously clarify the segmentation, mainly aiming for the maximum segmentation precision. This iterative approach allows the network to continuously enhance its capability to differentiate between the target objects and background elements, leading to better detection results. From the comparative analysis, it is seen that the proposed scheme improves the positive rate by 9.3%, and reduces detection time and false negatives by 9.39% and 13.2% for the different patterns observed.

Author Contributions: Conceptualization, J.S. and G.B.; methodology, R.; software; validation and writing—original draft preparation, J.S., G.B., R. and M.T.; supervision, M.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data can be obtained from the corresponding author on request.

Acknowledgments: The authors are grateful to the anonymous referees for their comments and suggestions. The authors are also grateful to the management and staff.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bi, H.; Zhang, C.; Wang, K.; Tong, J.; Zheng, F. Rethinking camouflaged object detection: Models and datasets. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 5708–5724. [CrossRef]
2. Zhang, C.; Wang, K.; Bi, H.; Liu, Z.; Yang, L. Camouflaged object detection via neighbor connection and hierarchical information transfer. *Comput. Vis. Image Underst.* **2022**, *221*, 103450. [CrossRef]
3. Mondal, A. Camouflage design, assessment and breaking techniques: A survey. *Multimed. Syst.* **2022**, *28*, 141–160. [CrossRef]
4. PullaRao, C.; Guruva Reddy, A.; Rama Rao, C.B. Camouflaged object detection for machine vision applications. *Int. J. Speech Technol.* **2020**, *23*, 327–335. [CrossRef]
5. Shen, Y.; Lin, W.; Wang, Z.; Li, J.; Sun, X.; Wu, X.; Huang, F. Rapid detection of camouflaged artificial target based on polarization imaging and deep learning. *IEEE Photonics J.* **2021**, *13*, 7800309. [CrossRef]
6. Ji, G.P.; Fan, D.P.; Chou, Y.C.; Dai, D.; Liniger, A.; Van Gool, L. Deep gradient learning for efficient camouflaged object detection. *Mach. Intell. Res.* **2023**, *20*, 92–108. [CrossRef]
7. Liu, Y.; Wang, C.Q.; Zhou, Y.J. Camouflaged people detection based on a semi-supervised search identification network. *Def. Technol.* **2023**, *21*, 176–183. [CrossRef]
8. Xu, X.; Chen, S.; Lv, X.; Wang, J.; Hu, X. Guided multi-scale refinement network for camouflaged object detection. *Multimed. Tools Appl.* **2023**, *82*, 5785–5801. [CrossRef] [PubMed]
9. Yuanying, G.; Chuntong, L.; Hongcai, L.; Zhongye, L. A camouflage target detection method based on local minimum difference constraints. *J. Syst. Eng. Electron.* **2023**, *34*, 696–705.
10. Zhai, Q.; Li, X.; Yang, F.; Jiao, Z.; Luo, P.; Cheng, H.; Liu, Z. Mgl: Mutual graph learning for camouflaged object detection. *IEEE Trans. Image Process.* **2022**, *32*, 1897–1910. [CrossRef] [PubMed]
11. Xu, X.; Zhu, M.; Yu, J.; Chen, S.; Hu, X.; Yang, Y. Boundary guidance network for camouflage object detection. *Image Vis. Comput.* **2021**, *114*, 104283. [CrossRef]
12. Liu, Y.; Zhang, K.; Zhao, Y.; Chen, H.; Liu, Q. Bi-RRNet: Bi-level recurrent refinement network for camouflaged object detection. *Pattern Recognit.* **2023**, *139*, 109514. [CrossRef]
13. Ji, G.P.; Zhu, L.; Zhuge, M.; Fu, K. Fast camouflaged object detection via edge-based reversible re-calibration network. *Pattern Recognit.* **2022**, *123*, 108414. [CrossRef]

14. Bi, H.; Tong, J.; Zhang, C.; Mo, D.; Wang, X. Camouflaged objects detection network via contradiction detection and feature aggregation. *Multimed. Tools. Appl.* **2023**, 1–19. [CrossRef]
15. Zhang, Q.; Ge, Y.; Zhang, C.; Bi, H. TPRNet: Camouflaged object detection via transformer-induced progressive refinement network. *Vis. Comput.* **2022**, *39*, 4593–4607. [CrossRef]
16. Xiao, J.; Chen, T.; Hu, X.; Zhang, G.; Wang, S. Boundary-guided context-aware network for camouflaged object detection. *Neural. Comput. Appl.* **2023**, *35*, 15075–15093. [CrossRef]
17. Xu, W.; Wang, Z.; You, X.; Zhang, C. Efficient fast convolution architectures for convolutional neural network. In Proceedings of the 2017 IEEE 12th International Conference on ASIC (ASICON), Guiyang, China, 25–28 October 2017; pp. 904–907.
18. Vijayan, R.; Mohler, G. Forecasting Retweet Count during Elections Using Graph Convolution Neural Networks. In Proceedings of the 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA), Turin, Italy, 1–3 October 2018; pp. 256–262.
19. Amrutha, B.R.; Bindu, K.R. Detecting Hate Speech in Tweets Using Different Deep Neural Network Architectures. In Proceedings of the 2019 International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, 15–17 May 2019; pp. 923–926.
20. Available online: https://www.kaggle.com/datasets/mikurules/aqua-test (accessed on 13 October 2023).

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.