# Camouflaged Object Detection using SINet and Wavelets

Luca Moresca, Nicholas Suozzi, Valerio Santini
Sapienza University of Rome
lucamoresca12@gmail.com, nicholassuozzi@gmail.com, valerio.santini.97@gmail.com

**Abstract**

The detection of camouflaged objects (COD) is a challenging problem in computer vision, as it requires the distinction of objects that blend seamlessly with their surroundings. Traditional detection methods, which rely solely on spatial features, often fail to solve this problem due to the high similarity between camouflaged objects and their backgrounds. This study proposes a different approach using SINet (Search Identification Network) in combination with wavelet transforms to improve detection accuracy. Inspired by Frequency-Spatial Entanglement Learning (FSEL), our method integrates frequency domain features through the Discrete Wavelet Transform (DWT), enabling a better representation of global and local image structures. This work demonstrates the effectiveness of incorporating the wavelet transform for COD and suggests future improvements.

## 1 Introduction

Camouflaged object detection (**COD**) represents a significant challenge in the field of computer vision, as it aims to identify objects that blend in with their surroundings. Unlike traditional object detection problems, where objects clearly stand out from the background, **COD** deals with situations where objects change their appearance, such as color, texture and shape, to make themselves indistinguishable from the environment. This high similarity between camouflaged objects and background makes discrimination based solely on spatial features, which focus on the local intensity of pixels and their positions, difficult. Methods that rely solely on these features may be ineffective due to the interference of complex backgrounds and local properties of spatial features. Therefore, it is crucial to develop approaches that go beyond the limitations of spatial features to obtain accurate **COD** results. Existing methods try to reduce the impact of pixel similarity by maximizing the ability to distinguish spatial features, but they often ignore the sensitivity and locality of features in the spatial domain, leading to suboptimal results.

## 2 Dataset

The COD10K [1] dataset was developed to advance the study of camouflaged object detection (COD), with the aim of identifying objects that blend seamlessly into their environment. This dataset comprises 10,000 images covering 78 categories of camouflaged objects in different natural scenes, including aquatic, bird, amphibian and terrestrial animals. Each image is annotated hierarchically with category information, bounding box, object level and instance level, as well as being labelled with attributes representing real-world challenges and alpha-matting.

Images were collected from various photography websites, mainly Flickr, with permission for academic use. Keywords used in the search included terms such as 'camouflaged animal', 'undetectable animal', 'camouflaged fish' and 'camouflaged butterfly'. Images from other sources such as Visual Hunt, Pixabay, Unsplash and Free-images, which offer public domain stock photos, were also included. To avoid selection bias, 3,000 images of salient objects from Flickr and 1,934 non-camouflaged images depicting background scenes such as forests, snow, prairies and the sea were also collected.

The annotations were crowdsourced and organised hierarchically according to category, bounding box, attributes and object/instance. Five super-classes and 69 sub-classes were created based on the data collected. Bounding boxes were annotated to facilitate the task of object proposal, and each camouflaged image is labelled with attributes representing typical challenges, such as occlusion and undefined boundaries. Object-level (5,069) and instance-level (5,930) masks were created, similar to the annotations of the COCO dataset.

The size of the objects varies from 0.01% to 80.74% of the image, with an average of 8.94%. The objects in COD10K are more difficult to detect than in other datasets due to the global/local contrast. In addition, the dataset has less centre bias than other datasets and includes a large number of Full HD 1080p images. The dataset is divided into 6,000 images for training and 4,000 for testing, randomly selected from each subclass.

The results of the experiments show that COD10K is the most challenging dataset, suitable for more complex scenes. It was used to develop and test the SINet (Search Identification Network) framework, which demonstrated competitive performance compared to other models. The COD10K dataset represents a valuable resource for research in the field of camouflaged object detection, offering a wide range of annotated images and realistic challenges.

Figure 1: Dataset samples

# 3 Reference work

In order to realise a system capable of recognising camouflaged objects, we analysed the work: 'Frequency-Spatial Entanglement Learning for Camouflaged Object Detection'. [2]. It introduces a novel method called Frequency-Spatial Entanglement Learning (**FSEL**) to address the challenge of Camouflaged Object Detection (COD). The detection of such objects represents a significant challenge in the field of computer vision, given the high similarity between camouflaged objects and their surroundings, which makes their identification particularly complex.

Current methodologies for COD have certain limitations. Many focus mainly on spatial features, being vulnerable in the presence of complex backgrounds and showing difficulties in distinguishing subtle variations. Some recent approaches have begun to incorporate frequency cues; however, they often neglect information between high and low frequencies or introduce unwanted background noise.

To overcome these limitations, the FSEL method proposes a combination of global frequency characteristics and local spatial characteristics in order to improve discriminative ability in the detection of camouflaged objects.

## 3.1 FSEL

Il nostro modello si basa sull'architettura FSEL, composta da 3 blocchi principali: Entanglement Transformer Blocks (ETB), Joint Domain Perception Module (JDPM) e Dual-domain Reverse Parser (DRP). Per capire la struttura e il funzionamento del nostro lavoro è necessario introdurre brevemente questi moduli che sono stati utilizzati anche da noi ma con l'introduzione di alcune differenze. Il problema degli attuali COD, come già detto, risiende nel considerare solamente features spaziali, e non dare peso a componenti frequenziali che risulatano essere estremamente utili per il rilevamento di oggetti mimetizzati. Per fare ciò nel modello FSEL, viene utilizzata la **Fast fourier Transform** operazione in grado di estrarre features frequenziali da un'immagine.

- **Entanglement Transformer Blocks (ETBs)**: These blocks are designed for representation learning. They use frequency self-attention to characterize the relationships between different frequency bands, while a feed-forward entanglement network facilitates the interaction between features of different domains through entanglement learning.
- **Joint Domain Perception Module (JDPM)**: This module serves for semantic enhancement by integrating multi-receptive information from frequency-spatial domains. Specifically, JDPM reconstructs multi-receptive information by introducing a frequency transformation in multi-scale features. It uses Atrous convolutions to capture spatial features.
- **Dual-domain Reverse Parser (DRP)**: This module is designed for feature integration in the frequency and spatial domains. The DRP optimizes and aggregates different information from multi-level features in both frequency and spatial domains.

The **FSEL** method combines global frequency features and local spatial features to optimize the initial input features and improve their discriminative ability. Frequency self-attention is used to obtain global frequency features by modeling correlations between frequency bands.

Entanglement learning between frequency and spatial features allows them to learn and collaborate for optimization.
The **JDPM** and **DRP** extend the applicability of global frequency features by optimizing input features and generating representations that incorporate both frequency and spatial information. Experiments conducted on three widely used datasets (**CAMO**, **COD10K** and **NC4K**) demonstrate the superiority of **FSEL** over 21 state-of-the-art **COD** methods through quantitative and qualitative comparisons. The source code for the project is publicly available. The paper highlights how traditional **COD** methods focus primarily on spatial features, which may be susceptible to interference from complex backgrounds and limited in their local properties.

Frequency features, generated via the Fourier transform, have been shown to have global features that are useful for understanding image content.
This work proposes that combining global frequency features with local spatial features can overcome the limitations of spatial features and improve the accuracy of **COD** results. Several recent methods have begun to incorporate frequency clues, but they often focus only on high- and low-frequency features, neglecting other information.
**FSEL** addresses these limitations through comprehensive analysis of frequency band interactions and entanglement learning between frequency and spatial features.

# 4 Wavelets and Fourier transform

In the field of signal analysis, Fourier and wavelet transforms represent two fundamental tools, each with distinctive characteristics that determine their applicability in different contexts.

The Fourier transform decomposes a signal into its sinusoidal components, offering a complete representation in the frequency domain. However, this methodology does not provide information on the temporal localisation of the different frequencies in the signal, making it less effective in the analysis of signals that are non-stationary or characterised by significant temporal variations.

In contrast, the wavelet transform allows a decomposition of the signal in both the time and frequency domains by using compact support basis functions known as wavelets. This feature allows for variable resolution: high in time for high frequencies and high in frequency for low frequencies. Consequently, wavelets are particularly suitable for the analysis of transient or discontinuous signals, offering a more detailed representation of local variations in the signal.

In terms of applications, the Fourier transform is traditionally preferred for the analysis of stationary signals, such as audio signals, where frequency components are predominant and persistent over time. On the other hand, wavelet transforms find use in areas such as image compression and processing, video compression and fingerprint analysis, where the ability to represent local details and discontinuities is crucial.

The choice between Fourier transform and wavelet transform depends on the specific characteristics of the signal under investigation and the objectives of the analysis. For signals with stable frequency components over time, the Fourier transform offers an effective representation. For signals with rapid temporal variations or pronounced local characteristics, the wavelet transform is the most suitable tool due to its ability to provide a simultaneous view in the time and frequency domain. [3]

In our project, we therefore chose to use wavelets, believing them to be more suitable for extracting frequency components from an image. The FFT decomposes the signal into sinusoids of fixed frequency, whereas the wavelet transform wavelet transform uses scalable functions that provide a representation with variable resolution, allowing a richer analysis in the spatial-frequency domain. frequency domain. The multiresolution property allows the wavelet transform to extract four differential frequency components when applied: **Low-Low** i.e. global structure (low frequencies), **Low-High** horizontal edges, **High-Low** vertical edges and finally **High-High** texture and fine details as shown in figure 2.

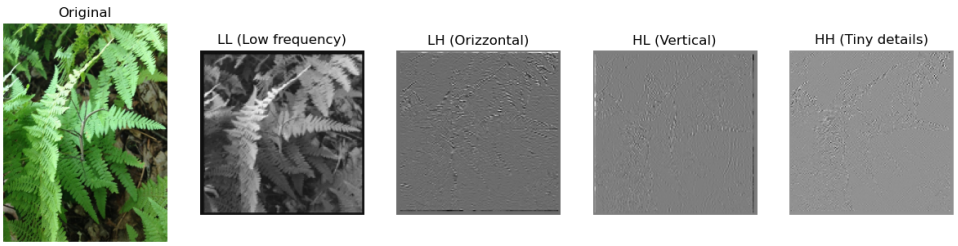


Figure 2: Wavelet application example

The wavelet transform used is as follows:

$$J_f \;=\; \Phi\|idwt(\sigma(dwt(J_S)) \cdot dwt(J_S))\|$$

This relation coincides exactly with the implementation of the Fourier transform, but the type of transformation used is changed. Otherwise, the implementation is the same using the coefficients extracted from the wavelet. The wavelet chosen was the **Haar** wavelet, as it is simple and quick to use, but may have slight loss problems on the fine edges. It would be interesting to test the implementation of "Coiflets", which have a higher level of detail accuracy but are computationally heavy. The advantages introduced by wavelets are mainly threefold: Better localisation than Fourier because it retains spatial and not just frequency detail. Multi-scale analysis allows for greater efficiency in detecting objects of different sizes; finally, compared to FFT, there is less sensitivity to noise and greater robustness to contrast variations.

# 5 Implementation

Taking a cue from the **FSEL**, we decided to create our own **CNN** that implements a **SINet** (Saliency Identification Network) for the detection of camouflaged objects, enriched with the use of **wavelet** transforms to improve the accuracy of the model.

## 5.1 SINet architecture

SINet, which stands for Search Identification Network, is a convolutional neural network designed to detect camouflaged objects, i.e. elements that blend in with their surroundings, making them difficult to detect. This model is inspired by the natural process of predation, which consists of two main phases: search and identification. Similarly, SINet consists of two key modules: the **Search Module** and the **Identification Module**. The former is responsible for the initial search for the camouflaged object within the image, while the latter focuses on the detailed identification of the object, refining the extracted features to precisely delineate its boundaries. SINet has demonstrated superior performance compared to various state-of-the-art object detection models on different datasets, making it a robust and general framework for the detection of camouflaged objects. Its applications range from biology, for studying animals with camouflage capabilities, to medicine, for detecting anomalies in diagnostic images, to the security industry, for detecting hidden threats [1].

## 5.2 Backbone

The network backbone is based on ResNet-50 for both our model and FSEL, a pre-trained architecture that is used to extract features from the input image. Processing takes place through five layers, each with increasing abstraction capacity. The first level applies an initial convolution followed by batch normalisation and a ReLU activation function to capture basic features such as edges and textures. Next, the image passes through a series of convolutional blocks, each of which extracts more complex information until the deepest level of the network is reached.

A fundamental difference to FSEL is the integration of the discrete wavelet transform (DWT), which is applied to the upper layers of the network. This transform decomposes each feature map into a low-frequency component and three high-frequency components. The low-frequency component Yl represents general image information, while the high-frequency components LH, HL and HH capture structural details such as intensity changes and texture variations.

After applying the transform, the high-frequency coefficients are rescaled by bilinear interpolation to maintain the same spatial dimension as the original feature map. The end result is an enriched representation that combines spatial and frequency information, improving the model's ability to distinguish camouflaged objects from their surroundings. These features are then passed to subsequent modules to refine the detection and segmentation.

## 5.3 Search Module

The Search Module has the task of identifying regions of the image that might contain a camouflaged object. To do so, it uses feature maps extracted from the intermediate layers of the ResNet. Each feature map is combined with its corresponding representation obtained via the discrete wavelet transform (DWT), which provides information on both global structures and high-frequency details.

Before being used, the frequency information is reduced in size through a 1x1 convolution, which retains only four meaningful channels for each feature level. This reduction helps to limit computational complexity and focus attention on the most relevant components for detection. After this step, each original feature map is concatenated with its respective compressed wavelet transform, resulting in a richer representation.

Subsequently, the resulting maps are processed by a series of 3x3 convolutions to extract more significant features. To ensure that all information is spatially aligned, maps from the deepest layers are rescaled by bilinear interpolation to be the same size. This multiscale fusion allows local details to be combined with broader contextual information, improving the model's ability to distinguish suspect regions.

The final result of this phase is a **coarse map**, a preliminary map that highlights the most likely areas where a camouflaged object might be located. This map, generated by means of a 1x1 convolution and normalised with a sigmoid function, represents an initial hypothesis on the location of hidden objects and is subsequently refined by the identification module to obtain a more precise segmentation.

## 5.4 Identification Module

The identification module has the task of refining the raw map generated in the search phase and obtaining a more precise segmentation of the camouflaged object. To do so, it uses features extracted from the deepest layer of the ResNet that contains high-level information about the image, combining them with its corresponding representation obtained via the discrete wavelet transform (DWT). The latter decomposes the map into low- and high-frequency components, allowing both structural and detailed information to be preserved.

Before being used, the x5 frequency components are processed by a 1x1 convolution that reduces the number of channels to four, providing a balance between computational efficiency and representation capability. Next, the map is concatenated with its compressed wavelet transform and processed by a 3x3 convolution to generate an initial refined representation. This is then upsampled with bilinear interpolation, increasing its spatial resolution to align it with the raw map generated in the previous step.

At this point, the coarse map is used as a guide to further improve the segmentation. It is concatenated with the refined map and passed through a new convolution, which allows the information from the two representations to be integrated. Finally, the result is processed by a final 1x1 convolution and normalised with a sigmoid function, producing the final segmentation map. This map represents the final estimate of the position and shape of the camouflaged object, providing a more precise result than the coarse.

## 5.5 SINet Module

The SINet class represents the main architecture of the model and is responsible for coordinating all steps in the process of detecting and segmenting the camouflaged objects. The first step takes place in the backbone, based on ResNet-50, which extracts the salient features of the image through a series of convolutional layers. In addition to traditional feature maps, frequency representations are also generated using the discrete wavelet transform (DWT), allowing details to be captured at different scales and frequencies.

Once the features have been extracted, the model moves on to the search phase, in which the search module analyses the intermediate layers of the network, combining spatial and frequency information to generate a coarse map. This map represents an initial hypothesis on the location of hidden objects in the image, highlighting regions that might contain camouflaged features.

Next, the identification phase refines the result using the identification module. This exploits features from the deeper layer of the ResNet, together with its wavelet representation, to refine the segmentation of the camouflaged object. The coarse map generated in the previous phase is used as a guide to improve the accuracy of the final result.

After identification, the model generates a segmented map that may have a lower resolution than the original image. To restore it to the correct size, the final output is rescaled by bilinear interpolation, ensuring that the segmentation maintains spatial consistency with the input image. At the end of the process, SINet returns both the final segmented map and the intermediate coarse map, providing a complete representation of the position and shape of the camouflaged objects within the scene.

## 5.6 Training

The training process of the SINet model is performed by iterating over a dataset of images and masks, with the objective of minimising the difference between the model predictions and the segmentation masks provided as ground truth. Both the training and validation phases are handled, using specific loss functions to optimise the quality of segmentation.

As Loss Function, we used a combination of **Dice Loss** and **Binary Cross Entropy**. The Dice Loss is a loss function based on the Dice coefficient, a metric that measures the overlap between the prediction and the ground truth. This loss is particularly useful in segmentation because it penalises predictions that do not overlap sufficiently with the correct mask, and has been implemented using the following relationship:

$$Dice \;=\; 2\,\frac{\sum\,(pred \cdot target) + \epsilon}{\sum pred \;+\; \sum target}$$

Where *pred* is the expected segmentation map and *target* is the ground truth. Consequently, the Dice Loss was calculated as: $Loss_{DIce} = 1 - Dice$

In addition, ECB Loss is a commonly used loss function for binary classification. It is used here because each pixel can be considered a binary classification (camouflaged object or background) and has been implemented according to the following relationship:

$$Loss_{BCE} = -\frac{1}{N}\sum(target \cdot log(pred) + (1 - target) \cdot (1 - pred))$$

Where *target* is the value of the ground truth mask and *pred* is the value of the prediction for the pixel. The overall loss is calculated according to the following relationship: $Loss = Loss_{BCE} + Loss_{Dice}$. The choice of these two functions was made because Dice loss improves the ability of Dice to segment camouflaged objects, while Binary Cross Entropy is needed to penalise individual wrong pixels.

The training was carried out for 180 epochs and with batch size 16.

# 6 Results

The results obtained demonstrate a very good performance in detecting objects. We will analyse three cases directly from the model output. The images are divided as follows: Original image, real mask, predicted mask.

## 6.1 Worst case

In this case, the images in figure 3 show a total absence of detection by the model. In this case, no mask has been predicted, which is an error because one can clearly see that the animals in the images are visible. There are several cases in which animals that are clearly visible to the eye are not detected.

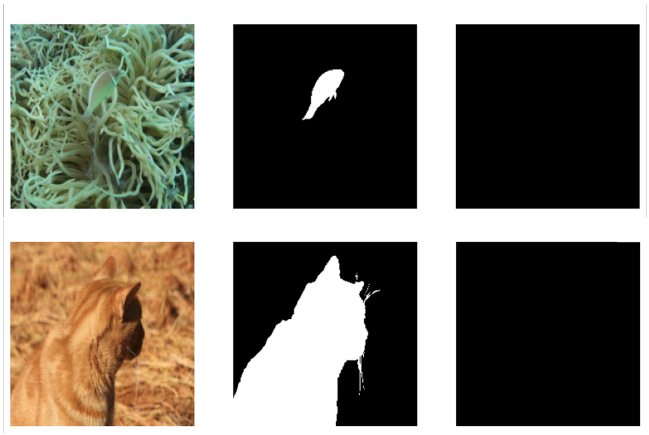

Figure 3: Worst detection case

## 6.2 Intermediate case

In the intermediate case, the model succeeds in detecting the presence of animals or camouflaged objects, only it fails to fully detect the figure, so the resulting mask is partially correct or has irrelevant details. For example In the second image in the figure 5 the presence of an animal is correctly detected but other details not present in the expected mask are detected.

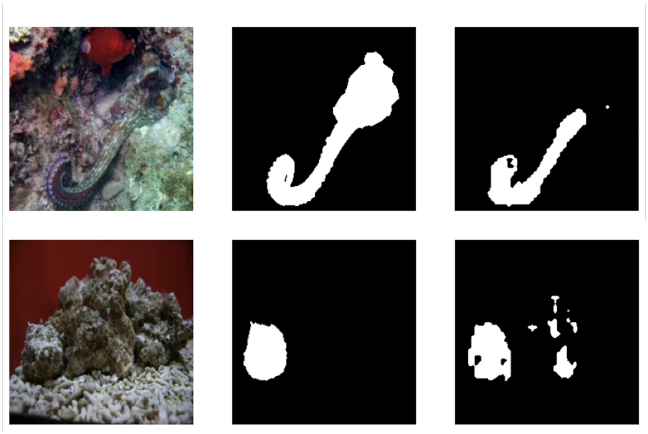

Figure 4: Intermediate detection case

## 6.3 Best case

In the best case the model succeeds in correctly and fully detecting the animal figure within the image, even if it is well camouflaged or partially covered by other objects. We could hardly achieve high levels of detail even in the best case, but the detection is to be considered more than optimal.

Figure 5: Best detection case

## 6.4 Evaluation

We chose four metrics to evaluate our model: Structural Similarity Measure, Enhanced Alignment Measure, WFM Weighted F-measure and Mean Absolute Error, which we will analyse in more detail.

### 6.4.1 S-Measure (Structural Similarity Measure)

S-Measure evaluates the structural similarity between model prediction and ground truth. This metric is designed to capture the structure of segmented objects, considering both the region of the object and the contrast with the background. The calculation was performed according to the following relationship and we obtained the result reported:

$$S = \alpha S_O + (1 - \alpha) S_\tau = 0.795$$

Where $S_O$ measures the similarity of the object between prediction and ground truth and $S_\tau$ measures the similarity of the background. The result obtained demonstrates the goodness of the model in detecting the overall structure of the segmented objects. Obviously, the value could be improved by increasing the model's edge definition.

### 6.4.2 E-Measure (Enhanced Alignment Measure)

E-Measure combines the accuracy of segmentation with the cohesion between predicted and actual pixels. It evaluates both the accuracy of the segmented object and the spatial alignment between prediction and ground truth:

$$E = \frac{1}{w \times h} \sum_{x=1}^{w} \sum_{y=1}^{h} (1 - |P(x,y) - G(x,y)|) = 0.965$$

Where $w$ and $h$ are the height and width of the image, $P(x,y)$ is the model prediction at pixel $(x,y)$ and $G(x,y)$ is the ground truth at pixel $(x,y)$ The result obtained is very good and indicates a strong alignment between the predicted segmentation and the ground truth. The model segments the areas containing camouflaged objects very accurately, capturing both their shape and position relative to the background well.

### 6.4.3 WFM (Weighted F-measure)

Weighted F-measure is a variant of the F1-score metric, which weights image areas unevenly, giving more weight to regions with more relevant information. This metric is useful when analysing images with complex backgrounds and objects of varying sizes, and was calculated with the following relationship:

$$F_\beta = \frac{(1+\beta^2) \cdot Precison \cdot Recall}{\beta^2 \cdot Precison + Recall} = 0.225$$

The low value obtained indicates that the model's performance is uneven across different images. The model has difficulties especially in scenarios where the object is very camouflaged or the contrast between object and background is low.

### 6.4.4 MAE (Mean Absolute Error)

The Mean Absolute Error (MAE) measures the mean absolute difference between the predicted segmentation and ground truth, providing an indication of the accuracy of the prediction at the pixel level.

$$E = \frac{1}{w \times h} \sum_{x=1}^{w} \sum_{y=1}^{h} |P(x,y) - G(x,y)| = 0.035$$

A value close to 0 indicates good segmentation, with minimal errors. A value of 0.035 indicates that the model has a low amount of pixel-wise errors. In combination with the low WFM, this indicates that the model is accurate in many cases, but fails completely in some specific scenarios.

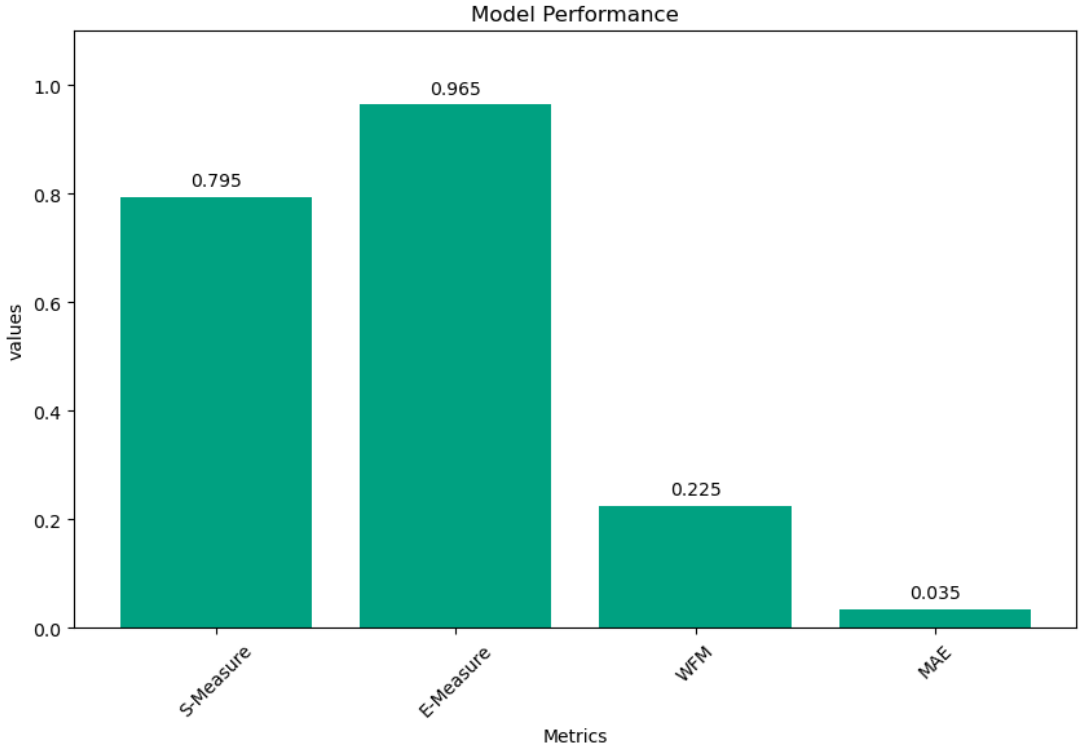


Figure 6: Model performance values

# 7 Conclusions

The identification of camouflaged objects represents a significant challenge in the field of computer vision, due to the strong similarity between the objects and their environment. In this study, we proposed an approach based on SINet (Search Identification Network) and the integration of the Discrete Wavelet Transform (DWT) to improve the model's ability to detect camouflaged objects. The use of wavelets made it possible to effectively combine spatial and frequency information, overcoming some of the limitations of traditional methods based solely on spatial features.

The results obtained show that the developed model is able to identify camouflaged objects accurately, especially in cases where the contrast between object and background is not extremely low. Evaluation metrics, including E-Measure and MAE, confirm that our approach is effective in the segmentation of hidden objects. However, some limitations were found, particularly in cases of extreme camouflage, where the lack of distinctive clues leads to false negatives.

In the future, the model could be improved by the adoption of more sophisticated wavelets, such as Coiflets, to further refine edge detection without excessively increasing the computational cost. Finally, further experiments on larger and more diverse datasets could provide further confirmation of the effectiveness of the proposed method.

# References


[1] Deng-Ping Fan et al. "Camouflaged object detection". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 2777–2787.

[2] Yanguang Sun et al. "Frequency-spatial entanglement learning for camouflaged object detection". In: *European Conference on Computer Vision*. Springer. 2024, pp. 343–360.

[3] Gilbert Strang. "Wavelet transforms versus Fourier transforms". In: *Bulletin of the American Mathematical Society* 28.2 (1993), pp. 288–305.