Neurocomputing 645 (2025) 130471

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

# Enhancing color selectivity in foundation models for downstream color vision tasks

Simone Bianco

*University of Milano-Bicocca, DISCo-Department of Informatics, Systems and Communication, Viale Sarca 336, Building U14, Milan, 20126, Italy*

ARTICLE INFO

Communicated by D. Liu

*Keywords:*
foundation models
DINOv2
Vision transformer
Color sensitivity
Color vision

ABSTRACT

The emergence of computer vision foundation models, inspired by the success of task-agnostic pretrained representations in Natural Language Processing (NLP), is revolutionizing the field. These models produce features that excel in downstream tasks even without fine-tuning. Last year, DINOv2 emerged, surpassing previous state-of-the-art general-purpose features on computer vision benchmarks, both at the image and pixel levels. In this work, we focus on what type of color information is embedded in DINOv2 features, and to assess their performance in computer vision tasks where color is a critical cue—for instance, recognizing the color of vehicles for traffic monitoring, detecting skin tones in biometric applications, or assessing product color attributes in fashion and e-commerce. Furthermore, we also propose a training-free feature transformation that increases color selectivity in DINOv2 features, i.e. their ability to respond differently to various colors in an image, boosting the performance on several classes of the color vision tasks considered.

## 1. Introduction

Foundation models in deep learning and artificial intelligence are large-scale, pre-trained models that serve as a general-purpose baseline for a wide range of downstream tasks [1]. These models are trained on extensive volumes of unlabeled data typically through self-supervised learning and are capable of capturing general-purpose knowledge, that can be fine-tuned or adapted for specific applications with relatively small task-specific datasets.

Computer vision foundation models have started to appear in 2021 [2] inspired by the success obtained in Natural Language Processing (NLP) by task-agnostic pretrained representations [3–6]. The main characteristic of these NLP foundation models is that they produce features that even without any further fine-tuning, are able to achieve performances on downstream tasks that are better than those produced by task-specific models [3]. Correspondingly, computer vision foundation models generate visual features that work well in any computer vision task, both when they are used for image-level tasks to describe a whole image (e.g., image classification tasks), both when they are used for region-level or pixel-level tasks to describe a portion of the image (e.g., image segmentation tasks).

In recent years several computer vision foundation models have started to appear like CLIP [7], DINO [8], ImageBind [9], SAM [10], and many others. In late 2023 DINOv2 appeared [11], which consists of a Vision Transformer (ViT) [12] model with one billion (1B) parameters distilled into a series of smaller models that surpass the previous best available general-purpose features [7] on most of the computer vision benchmarks at both image and pixel levels. Computer vision foundation models have been tested on many downstream tasks, also including other imaging domains, e.g. medical image classification [13]. Due to the increasing interest in computer vision foundation models, recent works started investigating some interesting properties, as for example biases [14].

In this work we want to assess the performance of DINOv2 features in computer vision tasks where color is a crucial information. The main contributions of this work are therefore:

- a thorough comparison of DINOv2 and ViT features on different categories of color computer vision tasks;
- an analysis of color selectivity for both DINOv2 and ViT features;
- a training-free transformation to boost the color selectivity of DINOv2 features.

## 2. Methodology

The focus of this paper is to test the performance of DINOv2 features on different color vision tasks. Since DINOv2 is a ViT model, two different types of features can be extracted from it:

- global features: the class token ([CLS]) feature having size $d$ is extracted. It serves to ViT models as a representation of an entire image, which can be used for classification.

*E-mail address:* simone.bianco@unimib.it.

https://doi.org/10.1016/j.neucom.2025.130471
Received 3 March 2025; Received in revised form 20 April 2025; Accepted 7 May 2025
Available online 23 May 2025

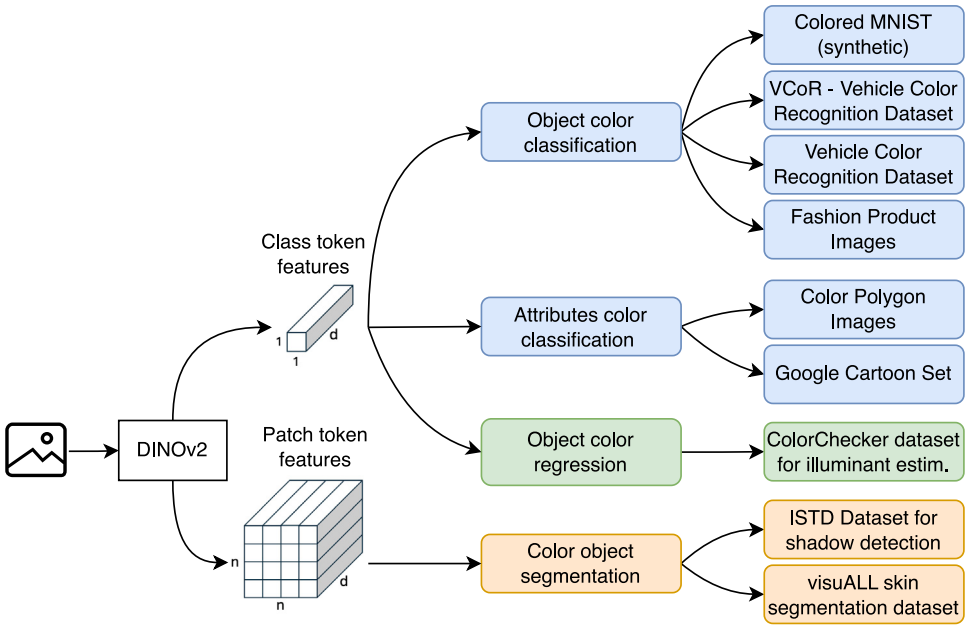


**Fig. 1.** Summary of the experimental setup with the type of features extracted for each task considered. The tasks considered are represented with distinct colors, and for each tasks the datasets considered are reported.

– local features: the patch token features having size $n \times n \times d$ are extracted. Since when an image is fed into a ViT it is divided into patches, these features provide a localized representation of the image.

DINOv2 comes in different sizes, which produce features with different dimensions: the smallest model DINOv2-S produces features having dimension $d = 384$; DINOv2-B produces features having dimension $d = 768$; the largest models, i.e. DINOv2-L and DINOv2-g produce features having dimension $d = 1024$ and $d = 1536$ respectively.

In order to test the performance of DINOv2 features on a variety of color vision tasks, four different classes of problems are considered:

– object color classification;
– attributes color classification (i.e., localized color classification);
– color object segmentation;
– object color regression (i.e., illuminant color estimation).

Within these classes of problems different datasets are selected. The main criterion for the selection of the datasets is that each of them must emphasize color as the primary discriminative feature. Among those satisfying this requirement, we performed the final selection providing different levels of difficulty, from synthetic (controlled) to real-world (noisy) data, and from coarse to fine-grained color tasks. A summary of the experimental setup is provided in Fig. 1.

DINOv2 features are compared with features coming from ViTs [12] trained through supervised learning in two different ways: (i) trained from scratch on Imagenet-1K (ILSVRC2012 [15]); (ii) with weights consisting of the original frozen Supervised Weakly through hashtAGs (SWAG) trunk weights [16] where the model has been weakly-supervised pre-trained using hashtag supervision, followed by a linear classifier learned on top of them trained on ImageNet-1K (denoted as SWAG-LIN in the experiments). In the tasks allowing it (i.e., classification and regression, where only image level features are considered), a further comparison is provided by features coming from a CNN architecture (i.e., ResNet-18 [17]) trained with supervised learning on Imagenet-1K; in the other task (i.e., segmentation) the further comparison is provided by methods in the state of the art.

In our experiments all the features are normalized to unitary norm, and a linear probe is trained on top of them. The choice of using a linear probe on top of frozen features is a very common approach [7,11,12] that allows testing how well the extracted features can support downstream tasks with a simple classifier, without fine-tuning the backbone model. Training only a linear probe on top of the extracted features is both fast and computationally efficient, and if it performs well on a downstream task, it suggests that the extracted features are highly informative.

In our experiments a single linear layer is trained for each of the object color classification tasks considered, mapping the $d$-dimensional features into the number of classes $c$ of each dataset considered, for a total of $c(d + 1)$ $(= d \cdot c + c)$ trainable parameters. For the object color regression task, a single linear layer is trained mapping the $d$-dimensional features into the three target values, for a total of $3(d + 1)$ $(= 3 \cdot d + 3)$ trainable parameters. For the color object segmentation task, a single convolutional layer with $1 \times 1$ filters is trained to preserve the spatial shape of the extracted features. The number of filters in the convolutional layer corresponds to the number of classes $c$ of each dataset considered, for a total of $c(d + 1)$ $(= d \cdot 1 \cdot 1 \cdot c + c)$ trainable parameters.

All the experiments are run in PyTorch on a single NVIDIA GeForce GTX 1080 GPU with 8 GB RAM, using Adam optimizer with a learning rate equal to $3 \cdot 10^{-4}$, a weight decay equal to $5 \cdot 10^{-4}$, a batch size of 16, and for a total of 200 epochs. The hyperparameters are selected by Bayesian optimization with Optuna by maximizing the classification performance on the CIFAR-100 dataset. This dataset is selected since it provides an upper bound in both the number of images and the number of classes considered in our experiments, thus assuring a proper convergence on all the color vision tasks considered.

## 3. Experimental results

In this section we report the experimental results on the four different classes of color vision problems considered.

### 3.1. Objects color classification

For this task four different datasets are considered. The first one is synthetically generated from the MNIST dataset [18] in order to have complete control on the color information added to the dataset as for example the distances among colors and the color selection schemes

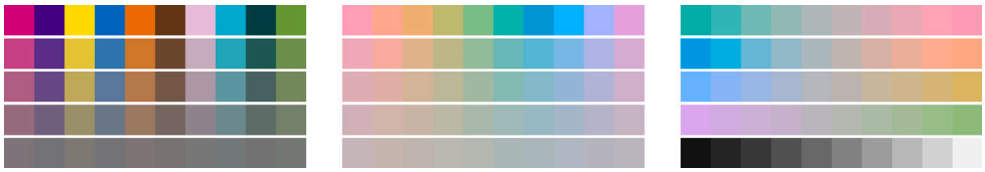

**Fig. 2.** Color palettes used to create the colored MNIST synthetic datasets: `High contrast MNIST` (left), `Hue circle MNIST` (middle), `Opponent directions MNIST` (right).

to carefully test the capabilities of DINOv2 models in discriminating different colors. In all the experiments, a total of ten different colors are considered, and each MNIST sample is randomly assigned to one color class. The original partition of MNIST is considered, with 60,000 training images and 10,000 test images.

Let us denote $S_i$ the $i$th gray level MNIST sample and let $j$ be the color class randomly assigned to it, with associated color $[R_j, G_j, B_j]$. The colorized sample is then created as:

$$[R_i, G_i, B_i] = \left[\frac{S_i}{\max S_i}, \frac{S_i}{\max S_i}, \frac{S_i}{\max S_i}\right] \cdot \begin{bmatrix} R_j & & \\ & G_j & \\ & & B_j \end{bmatrix} \quad (1)$$

i.e., the gray level image is scaled by its maximum value, then it is replicated three times along the channel dimension, and finally a von Kries-like diagonal transformation [19] is applied. Three main color MNIST datasets are generated, differing in how the ten possible colors are selected:

`High contrast`: the palette of ten colors is selected so that minimum color difference among all possible pair of colors in the palette is maximized. This results in a set of colors having a minimum distance of 34 $\Delta E$ units in the CIELab uniform color space [20]. The rationale behind this color scheme is that it ensures the largest perceptual differences between colors, enabling a straightforward evaluation of a model's capacity to discriminate distinct hues. Other four versions of this palette are generated by mapping the colors in CIELab color space and uniformly scaling their coordinates towards the average gray. This results in four additional palettes having a minimum distance of 26, 18, 10, and 2 $\Delta E$ units respectively.

`Hue circle`: the palette of ten colors is selected in CIELab space on the $L = 75$ plane at ten constant, evenly distributed, hue angles starting at 0°. It is selected at the maximum saturation that allows the circle to be completely contained in the CIELab color gamut, resulting in a difference of 15 $\Delta E$ units among colors. Other four versions of this palette are generated by reducing the saturation, resulting in a difference of 12, 9, 6, and 3 $\Delta E$ units respectively. The rationale behind this color scheme is that a hue circle in a uniform color space like CIELab ensures equal spacing in terms of hue, enabling a comprehensive test across the color spectrum.

`Opponent`: five different palettes are created along five different opponent directions passing through the origin of the $L = 75$ plane in CIELab color space. Four different hue angles are considered, evenly spaced, starting from 0° and resulting in a difference of 10 $\Delta E$ units between consecutive colors in each palette. The fifth palette is instead sampled along the achromatic axis, keeping the same $\Delta E$ distance among the chosen colors. The rationale behind this color scheme is that it is biologically inspired from the opponent process theory of color vision (e.g., red–green and blue–yellow axes). These directions align with how human vision processes and differentiates colors. The 15 palettes obtained are depicted in Fig. 2, while the performance of the methods tested is respectively reported in Tables 1, 2, and 3.

From the results reported we can observe how in most of the cases DINOv2 features are better than ResNet-18 features, with the only exception being DINOv2-S on the set-1 of `Hue circle MNIST`. ViT features instead reach the highest recognition accuracy in all the different configurations, with DINOv2 features being close on the `High contrast MNIST dataset` (3.3% drop on average). This gap increases on the `Hue circle MNIST` and `Opponent directions MNIST` datasets, where the average drop in performance is about 12.4%. In general, we observe that accuracy decreases as the color differences among the palette colors reduces (i.e., in `High contrast MNIST` and `Hue circle MNIST` from set-1 to set-5), mirroring a phenomenon observed in the human visual system, where the ability to distinguish colors also declines for very similar hues or tones. We can also observe how there are opponent directions along which the difference in performance is significantly larger than in others (e.g., set-1 and set-4 in `Opponent directions MNIST`).

The second dataset considered is the VCor - Vehicle Color Recognition dataset [21],[1] where the task is to classify vehicle colors into 15 different classes. The dataset contains 7267 training images and 1556 test images. Some examples of images within this dataset are depicted in Fig. 3. The recognition results are reported in Table 4(a), where it can be seen that ViT features obtain the highest accuracy, followed by ResNet-18 features with an average drop in performance of about 22.4%, followed by DINOv2 features with an accuracy that is on average 31.7% less than that of ViT features. In particular we can also observe how the performance of DINOv2 features tends to reduce with the increase of the model size.

The third dataset considered is also related to vehicle color recognition [22], where now the task is to classify vehicle colors into 8 different classes. The dataset has 7802 training images and 7799 test images. Some examples of images within this dataset are shown in Fig. 4. The recognition results are reported in Table 4(b), where it can be seen that ViT features obtain the highest accuracy, followed by DINOv2 features with an average drop in performance of about 8.2%, followed by ResNet-18 features with an accuracy that is on average 12.7% less than that of ViT features. As for the previous dataset we can observe that the performance of DINOv2 features tends to reduce with the increase of the model size.

The fourth dataset considered is related to fashion product color recognition on the `Fashion Product Images` dataset,[2] where now the task is to classify fashion product colors into 47 different classes. The dataset has 35,554 training images and 8887 test images. Some sample images within this dataset are shown in Fig. 5. The recognition results are reported in Table 4(c), where it can be seen that ViT features obtain the highest accuracy, followed by DINOv2 and ResNet-18 features with an average drop in performance of about 8.6%. Also for this dataset we can observe that the performance of DINOv2 features tends to reduce with the increase of the model size.

### 3.2. Localized attributes color classification

In this subsection two different datasets are considered. The first one is the `Color Polygon Images` dataset,[3] consisting of images of colored polygons against a colored background. In this dataset the task is to classify the foreground (i.e., polygon) and background colors

[1] https://www.kaggle.com/datasets/landrykezebou/vcor-vehicle-color-recognition-dataset.

[2] https://www.kaggle.com/datasets/paramaggarwal/fashion-product-images-small.

[3] https://www.kaggle.com/datasets/gonzalorecioc/color-polygon-images.

**Table 1**
Accuracy results on the synthetic `High contrast MNIST` dataset.

| Method | Set-1 34 $\Delta E$ Acc. (%) ↑ | Set-2 26 $\Delta E$ Acc. (%) ↑ | Set-3 18 $\Delta E$ Acc. (%) ↑ | Set-4 (10 $\Delta E$) Acc. (%) ↑ | Set-5 (2 $\Delta E$) Acc. (%) ↑ |
|---|---|---|---|---|---|
| ResNet-18 | 95.9 | 92.8 | 82.6 | 58.8 | 17.9 |
| ViT-B/16 | 99.8 | 99.2 | 97.9 | 92.5 | 40.7 |
| ViT-B/16 (SWAG-LIN) | **100.0** | **100.0** | 99.9 | 99.1 | 55.2 |
| ViT-L/16 | **100.0** | 99.7 | 99.6 | **99.7** | 64.6 |
| ViT-L/16 (SWAG-LIN) | **100.0** | 99.9 | 99.9 | 98.7 | 55.1 |
| ViT-H/14 (SWAG-LIN) | **100.0** | **100.0** | **100.0** | **99.7** | **72.7** |
| DINOv2 ViT-S/14 | 96.8 | 94.9 | 94.1 | 89.4 | 36.6 |
| DINOv2 ViT-B/14 | 96.2 | 95.0 | 94.6 | 91.1 | 51.5 |
| DINOv2 ViT-L/14 | 96.4 | 95.8 | 95.4 | 93.5 | 56.2 |
| DINOv2 ViT-g/14 | 96.2 | 96.7 | 97.8 | 97.2 | 68.9 |

**Table 2**
Accuracy results on the synthetic `Hue circle MNIST` dataset.

| Method | Set-1 34 $\Delta E$ Acc. (%) ↑ | Set-2 26 $\Delta E$ Acc. (%) ↑ | Set-3 18 $\Delta E$ Acc. (%) ↑ | Set-4 (10 $\Delta E$) Acc. (%) ↑ | Set-5 (2 $\Delta E$) Acc. (%) ↑ |
|---|---|---|---|---|---|
| ResNet-18 | 85.4 | 70.6 | 56.9 | 39.0 | 20.8 |
| ViT-B/16 | 98.1 | 94.1 | 89.2 | 77.6 | 48.2 |
| ViT-B/16 (SWAG-LIN) | 99.9 | 99.7 | 98.8 | 94.6 | 75.9 |
| ViT-L/16 | **100.0** | **99.9** | **100.0** | **99.4** | 80.6 |
| ViT-L/16 (SWAG-LIN) | 99.9 | 99.7 | 99.5 | 96.3 | 78.4 |
| ViT-H/14 (SWAG-LIN) | **100.0** | **99.9** | 99.7 | 98.7 | **90.0** |
| DINOv2 ViT-S/14 | 84.1 | 75.5 | 72.4 | 62.5 | 41.1 |
| DINOv2 ViT-B/14 | 93.7 | 88.2 | 85.6 | 80.3 | 56.1 |
| DINOv2 ViT-L/14 | 92.1 | 87.6 | 85.8 | 82.8 | 62.6 |
| DINOv2 ViT-g/14 | 96.0 | 91.4 | 92.0 | 90.0 | 76.7 |

**Table 3**
Accuracy results on the synthetic `Opponent directions MNIST` dataset.

| Method | Set-1 (0°) Acc. (%) ↑ | Set-2 (45°) Acc. (%) ↑ | Set-3 (90°) Acc. (%) ↑ | Set-4 (135°) Acc. (%) ↑ | Set-5 (achrom.) Acc. (%) ↑ |
|---|---|---|---|---|---|
| ResNet-18 | 65.2 | 65.1 | 44.5 | 51.6 | 46.5 |
| ViT-B/16 | 76.5 | 79.0 | 72.8 | 70.6 | 65.7 |
| ViT-B/16 (SWAG-LIN) | 89.5 | 93.9 | 94.6 | 89.7 | 67.2 |
| ViT-L/16 | 86.0 | 90.7 | 89.5 | 80.5 | 56.1 |
| ViT-L/16 (SWAG-LIN) | 88.3 | 92.6 | 96.1 | 88.5 | 66.5 |
| ViT-H/14 (SWAG-LIN) | **91.9** | **96.2** | **97.4** | **92.4** | **76.2** |
| DINOv2 ViT-S/14 | 67.2 | 79.1 | 73.1 | 61.6 | 52.2 |
| DINOv2 ViT-B/14 | 76.0 | 86.8 | 80.5 | 70.8 | 54.4 |
| DINOv2 ViT-L/14 | 71.4 | 80.0 | 77.1 | 70.2 | 57.0 |
| DINOv2 ViT-g/14 | 76.8 | 87.3 | 83.0 | 74.8 | 54.8 |

**Fig. 3.** Examples from the `VCoR – Vehicle Color Recognition Dataset` [21].

**Fig. 4.** Examples from the vehicle color recognition dataset.

**Fig. 5.** Examples from the `Fashion Product Images` dataset.

**Table 4**
Accuracy results on the object color classification tasks: (a) on the VCoR – Vehicle Color Recognition Dataset [21] (15 classes); (b) on the vehicle color recognition dataset [22] (8 classes); (c) on the Fashion Product Images dataset (47 classes).

| Method | (a) Acc. (%) ↑ | (b) Acc. (%) ↑ | (c) Acc. (%) ↑ |
|---|---|---|---|
| ResNet-18 | 54.6 | 70.9 | 42.7 |
| ViT-B/16 | 71.9 | 85.8 | 48.9 |
| ViT-B/16 (SWAG-LIN) | 76.5 | 81.3 | 51.3 |
| ViT-L/16 | 78.8 | **88.5** | 53.8 |
| ViT-L/16 (SWAG-LIN) | **79.0** | 83.4 | 51.5 |
| ViT-H/14 (SWAG-LIN) | **79.0** | 78.9 | 51.1 |
| DINOv2 ViT-S/14 | 51.8 | 79.5 | 44.9 |
| DINOv2 ViT-B/14 | 46.1 | 75.1 | 42.9 |
| DINOv2 ViT-L/14 | 41.6 | 75.1 | 41.1 |
| DINOv2 ViT-g/14 | 42.0 | 72.0 | 41.8 |
| DINOv2 ViT-S/14 ColSelBoost | 61.8 | 85.7 | 51.7 |
| DINOv2 ViT-B/14 ColSelBoost | 58.8 | 82.2 | 50.4 |
| DINOv2 ViT-L/14 ColSelBoost | 53.5 | 81.4 | 49.7 |
| DINOv2 ViT-g/14 ColSelBoost | 60.3 | 79.5 | **54.1** |

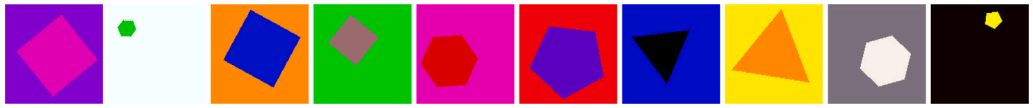


**Fig. 6.** Examples from the Color Polygon Images. Labels associated for foreground (FG) and background (BG) color classification: (a) FG:0 BG:7; (b) FG:1 BG:1; (c) FG:2 BG:4; (d) FG:3 BG:0; (e) FG:4 BG:5; (f) FG:5 BG:9; (g) FG:6 BG:8; (h) FG:7 BG:2; (i) FG:8 BG:6; (j) FG:9 BG:3.

**Table 5**
Accuracy results on the localized attribute color classification task: (a) on the Color Polygon Images dataset (10 classes for both foreground and background color classification); (b) on the Google Cartoon Set (5 classes for eye, 10 for hair, and 11 for skin color classification).

| Method | (a) | | (b) | | |
|---|---|---|---|---|---|
| | Foregr. Acc. (%) ↑ | Backgr. Acc. (%) ↑ | Eye Acc. (%) ↑ | Hair Acc. (%) ↑ | Skin Acc. (%) ↑ |
| ResNet-18 | 74.4 | 97.3 | 59.7 | 80.0 | 32.5 |
| ViT-B/16 | 88.7 | 97.5 | 79.5 | 93.4 | 59.0 |
| ViT-B/16 (SWAG-LIN) | 89.4 | 96.4 | 75.4 | 92.7 | 61.8 |
| ViT-L/16 | 94.0 | 98.5 | 80.3 | 95.9 | 77.6 |
| ViT-L/16 (SWAG-LIN) | 87.5 | 96.4 | 80.4 | 92.8 | 65.5 |
| ViT-H/14 (SWAG-LIN) | 90.8 | 95.7 | 79.4 | 94.8 | 68.2 |
| DINOv2 ViT-S/14 | 74.8 | 91.8 | 71.5 | 69.7 | 31.7 |
| DINOv2 ViT-B/14 | 77.7 | 92.1 | 81.3 | 72.6 | 35.4 |
| DINOv2 ViT-L/14 | 82.0 | 93.5 | 79.4 | 81.3 | 39.3 |
| DINOv2 ViT-g/14 | 83.2 | 91.3 | 80.7 | 89.5 | 46.0 |
| DINOv2 ViT-S/14 ColSelBoost | 87.4 | 96.4 | 74.8 | 77.2 | 39.1 |
| DINOv2 ViT-B/14 ColSelBoost | 89.7 | 96.3 | 81.9 | 83.9 | 48.7 |
| DINOv2 ViT-L/14 ColSelBoost | 91.6 | 97.3 | 81.5 | 90.1 | 56.3 |
| DINOv2 ViT-g/14 ColSelBoost | **96.6** | **98.8** | **82.3** | **96.2** | **71.2** |

in ten different classes. Some example images within this dataset are reported in Fig. 6. The dataset contains 7000 training images and 3000 test images. The recognition results are reported in Table 5(a), where it can be seen that on the foreground color classification task ViT features obtain the highest accuracy, followed by DINOv2 and ResNet-18 features with an average drop in performance of about 10.7% and 15.7% respectively. On this task we can also observe that the performance of DINOv2 features tends to increase with the increase of the model size.

Concerning the background color classification task instead, overall we can observe accuracy that are on average 8.9% higher than those for foreground color classification task. We can also observe how ResNet-18 features obtain the highest accuracy, followed by ViT features with an average drop in performance of about 0.4%, followed by DINOv2 features with an accuracy that is on average 5.1% less than that of ResNet-18 features. On this task we can observe that the best performance is obtained by DINOv2-B and DINOv2-L features.

The second dataset considered is the Google Cartoon Set,[4] consisting of images of cartoon faces. In this dataset the task is to classify the color of the eyes in 5 different classes, the hair color in 10 different classes and the skin color in 11 different classes. The dataset contains 10,000 training images and 10,000 test images. Some examples taken from this dataset are reported in Fig. 7, while the recognition results are reported in Table 5(b).

On the eye color classification task ViT features obtain the highest accuracy, followed by DINOv2 features with an average drop in performance of about 1.7%, and followed by ResNet-18 features with an average accuracy that is about 20.2% less than that of ViT features. Concerning the hair color classification task ViT features obtain the highest accuracy, followed by ResNet-18 features with an average drop

[4] https://google.github.io/cartoonset/download.html.

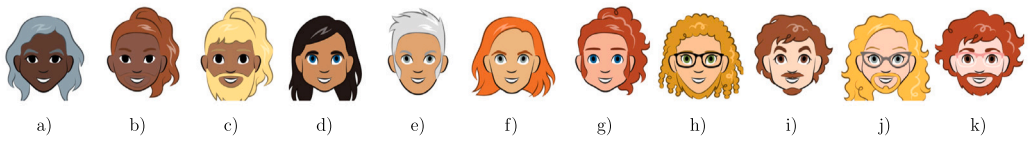


**Fig. 7.** Examples from the Google Cartoon Set. Labels associated for the eye (E), hair (H), and skin (S) color classification: (a) E:4 H:8 S:0; (b) E:4 H:5 S:1; (c) E:4 H:0 S:2; (d) E:1 H:7 S:3; (e) E:3 H:9 S:4; (f) E:3 H:2 S:5; (g) E:1 H:3 S:6; (h) E:2 H:4 S:7; (i) E:0 H:5 S:8; (j) E:3 H:1 S:9; (k) E:0 H:3 S:10.

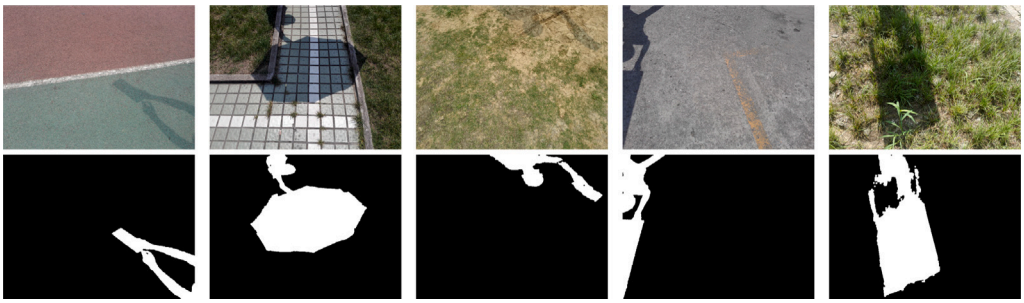

**Fig. 8.** Example from the ISTD Dataset: color images (top row); shadow segmentation ground truths (bottom row).

in performance of about 13.9%, and followed by DINOv2 features with an average accuracy that is about 15.6% less than that of ViT features. Finally, on the skin color classification task ViT features obtain the highest accuracy, followed by DINOv2 features with an average drop in performance of about 28.3%, and followed by ResNet-18 features with an average accuracy that is about 33.9% less than that of ViT features. On this dataset we can observe that the performance of DINOv2 features tends to increase with the increase of the model size for the hair and skin color classification tasks, while on the eye classification task DINOv2-B is the one reaching the highest accuracy.

### 3.3. Color object segmentation

In this subsection two different datasets are considered. Since ResNet-18 cannot be used for image segmentation since a single feature is computed for the whole input image, ViT and DINOv2 features are compared with state of the art algorithms.

The first task considered is shadow detection on the ISTD Dataset [23]. The dataset has 1330 training images and 540 test images, all having a fixed size of 640 × 480. In case of DINOv2 features, these are resized to 224 × 224 which results in 16 × 16 maps. For training, these maps are upsampled to 320 × 240 with bilinear upsampling, while for testing they are upsampled in the same way to 640 × 480. In case of ViT features, input images are resized to 256 × 256 for ViT-B and ViT-L, and to 224 × 224 for ViT-L to ensure that in all cases the feature map is 16 × 16. Some sample images within this dataset are provided in Fig. 8 together with the corresponding ground truths. The results are reported in Table 6 in terms of Balanced Error Rate (BER), which is defined as the average between False Positive Rate and False Negative Rate. We can observe how DINOv2 features obtain errors that are much lower than those of ViT features, which are on average 2.05× higher. Compared with methods in the state of the art we can see that DINOv2-S, -B and -L obtain results better than cGAN [24], and similar to StackedCNN [25]. The larger DINOV2-g instead is better than both cGAN [24] and StackedCNN [25], with a BER that is close to that of scGAN [24]. These results are noteworthy since we recall that DINOv2 predicts a 16 × 16 map that is upsampled to the groundtruth size (with an upsampling factor of about 35×), while state-of-the-art methods obtain these results working on full resolution images.

The second task considered is skin segmentation on the visuAAL skin segmentation dataset [26]). This dataset contains images of different aspect ratios having the longest side equal to 1024 pixels, for a total of 3000 training images and 1157 test images. For ViT and DINOv2 features input images are resized in the same way of the previous dataset to estimate a 16 × 16 map. This map is then upsampled to 224 × 224 with bilinear interpolation for training, while in testing it is upsampled in the same way to the original size of the ground truth. Some sample images within this dataset are provided in Fig. 9 together with the corresponding ground truths. The results are reported in terms of Precision, Recall, and F1-score in Table 7. We can observe how DINOv2 features obtain scores that are much higher than those of ViT features for all the metrics considered (i.e., on average 18.0% higher). Compared with methods in the state of the art we can see that DINOv2-B and DINOv2-L obtain scores close to FCN [27] and HLNet [30] (i.e., on average 3.7% lower). As for the previous dataset, these results are noteworthy since we have to recall that state-of-the-art methods obtain these results working on full resolution images, while DINOv2 predicts a 16 × 16 maps that are upsampled to the groundtruth size with an upsampling factor that is on average 52×, due to the high resolution images present in the dataset. These results could be further improved using larger image input size, model-agnostic features upsampling [32], or multi-scale feature representations [33]. However, such enhancements were not explored in this study in order to maintain a consistent experimental setup across all evaluated tasks.

**Table 6**
Performance in terms of Balanced Error Rate (BER) for the shadow detection task on the ISTD Dataset [23].

| Method | BER (%) ↓ |
|---|---|
| StackedCNN [25] | 8.6 |
| cGAN [24] | 9.6 |
| scGAN [24] | 4.7 |
| ST-CGAN [23] | **3.9** |
| ViT-B/16 | 8.4 |
| ViT-B/16 (SWAG-LIN) | 16.2 |
| ViT-L/16 | 6.8 |
| ViT-L/16 (SWAG-LIN) | 21.6 |
| ViT-H/14 (SWAG-LIN) | 28.4 |
| DINOv2 ViT-S/14 | 8.8 |
| DINOv2 ViT-B/14 | 8.7 |
| DINOv2 ViT-L/14 | 8.6 |
| DINOv2 ViT-g/14 | 5.6 |
| DINOv2 ViT-S/14 ColSelBoost | 9.2 |
| DINOv2 ViT-B/14 ColSelBoost | 9.3 |
| DINOv2 ViT-B/14 ColSelBoost | 9.3 |
| DINOv2 ViT-g/14 ColSelBoost | 5.6 |

**Fig. 9.** Examples from the visuAAL skin segmentation dataset: color images (top row); skin segmentation ground truths (bottom row).

**Table 7**
Performance in terms of precision, recall and F1-score for the skin segmentation task on the visuAAL skin segmentation dataset [26].

| Method | Precision (%) ↑ | Recall (%) ↑ | F1-score (%) ↑ |
|---|---|---|---|
| FCN [27] | 70.34 | 84.88 | 76.80 |
| SegNet [28] | 80.71 | 80.12 | 80.29 |
| UNet [29] | **82.66** | **85.34** | 83.83 |
| HLNet [30] | 76.50 | 79.86 | 78.01 |
| DSNet [31] | 85.80 | 85.08 | **85.40** |
| ViT-B/16 | 56.82 | 66.46 | 61.27 |
| ViT-B/16 (SWAG-LIN) | 55.23 | 66.36 | 60.28 |
| ViT-L/16 | 59.62 | 65.44 | 62.39 |
| ViT-L/16 (SWAG-LIN) | 50.37 | 63.53 | 56.19 |
| ViT-H/14 (SWAG-LIN) | 34.87 | 33.05 | 33.94 |
| DINOv2 ViT-S/14 | 69.41 | 74.94 | 72.07 |
| DINOv2 ViT-B/14 | 70.08 | 76.68 | 73.23 |
| DINOv2 ViT-L/14 | 70.77 | 73.20 | 71.97 |
| DINOv2 ViT-g/14 | 72.71 | 76.67 | 74.64 |
| DINOv2 ViT-S/14 ColSelBoost | 69.71 | 74.20 | 71.88 |
| DINOv2 ViT-B/14 ColSelBoost | 70.37 | 75.13 | 72.67 |
| DINOv2 ViT-L/14 ColSelBoost | 70.67 | 72.15 | 71.40 |
| DINOv2 ViT-L/14 ColSelBoost | 72.36 | 76.06 | 74.16 |

### 3.4. Object color regression

In this subsection we consider the task of illuminant color regression on the ColorChecker [34] dataset. This task is usually referred to as illuminant estimation or computational color constancy, which substantially consists of regressing the color of the scene illuminant from the image content alone. This task is essential in both digital photography and computer vision, particularly for downstream applications (e.g., [35]) where estimating and correcting for scene lighting conditions is necessary to achieve illumination-invariant representations [36]. The dataset has 568 images split into three folds used to train the models in a 3-fold cross-validation setup. Some sample images within this dataset with the corresponding ground truths are depicted in Fig. 10. Regression results in terms of angular error statistics between the estimated and ground truth illuminant colors are reported in Table 8, where the performance of several algorithms in the state of the art is also reported.

Among the statistics reported, the median error is usually the most appropriate measure to compare the algorithms [37]. On this task we can observe that ViT features obtain the lowest median error, followed by DINOv2 features, with an average median error that is 1.09× that of ViT features. ResNet-18 features instead obtain the highest median error, that is on average 1.19× the one obtained by ViT features. Compared with the methods in the state of the art, the performance of ViT and DINOv2 features is similar to that of statistics-based methods (e.g., SoG [38], GGW [38], GE1 [38], and GE2 [38]), while deep learning-based methods trained end-to-end (e.g., FFCC [39], FC$^4$ [40], Conv. Mean [41], QU [42], and QU+ft [42]) obtain much lower errors. These results are expected, since it is known that this type of problem needs a full training on a proper dataset or even a proper network architecture to obtain an accurate illuminant estimate [43,44]. This is due to the fact that in color constancy the features should be able to estimate a property of the image (i.e., the illuminant color) that is not directly visible in the image, but we can only observe its interaction with the surfaces of the objects in the scene. It is interesting to note how ViT-L features obtain a maximum error that is the lowest among all the methods considered.

Summarizing the key trends observed across the investigated color vision tasks, we can notice how supervised ViTs outperform self-supervised DINOv2 in image-level tasks (i.e., classification and regression): ViT features (especially SWAG-LIN variants) consistently achieved higher accuracy in object color classification and lower angular error in illuminant estimation. In particular, we can observe that the performance difference does not depend on the number of color classes in each dataset. In pixel-level tasks (i.e. segmentation), DINOv2 features outperform ViT-based models: this suggests that self-supervised learning in DINOv2 captures robust localized color features, making it more effective for spatially structured tasks.

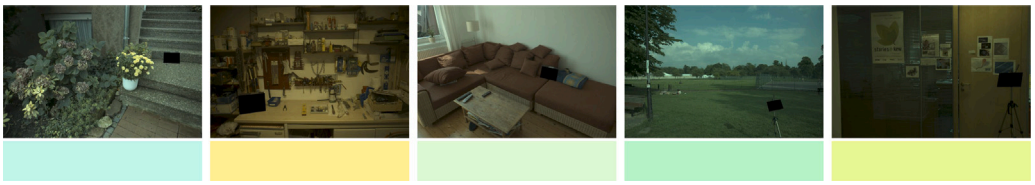

**Fig. 10.** Examples from the ColorChecker dataset: color images (top row); ground truth illuminant color (bottom row).

**Table 8**
Illuminant color estimation performance in terms of angular error on the ColorChecker dataset [34].

| Method | Angular error ↓ | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Mean | Med. | Tri-m. | B-25 | W-25 | 95-P | 99-P | Max |
| SoG [38] | 4.07 | 2.70 | 3.14 | 0.55 | 9.79 | 12.20 | 17.00 | 21.89 |
| GGW [38] | 4.05 | 2.60 | 3.06 | 0.56 | 9.78 | 12.30 | 16.97 | 21.16 |
| GE1 [38] | 3.89 | 2.77 | 3.10 | 0.78 | 8.83 | 11.09 | 14.57 | 22.60 |
| GE2 [38] | 3.89 | 2.88 | 3.13 | 0.75 | 8.80 | 10.90 | 14.02 | 23.10 |
| FFCC (model J) [39] | 2.23 | 1.45 | 1.59 | **0.35** | 5.46 | 7.33 | **10.85** | 17.27 |
| $FC^4$ [40] | **2.14** | **1.44** | **1.57** | 0.40 | **5.08** | **6.50** | 12.75 | **15.28** |
| Conv. Mean [41] | 2.50 | 1.73 | 1.90 | 0.51 | 5.81 | 7.93 | 12.42 | 16.13 |
| QU [42] | 3.26 | 2.07 | 2.38 | 0.44 | 7.98 | 10.78 | 14.32 | 21.68 |
| QU+ft [42] | 3.02 | 2.07 | 2.26 | 0.50 | 7.15 | 9.22 | 12.92 | 17.05 |
| ResNet-18 | 4.25 | 2.89 | 3.25 | 0.78 | 9.87 | 12.77 | 16.64 | 20.10 |
| ViT-B/16 | 3.71 | 2.74 | 2.96 | 0.83 | 8.40 | 10.65 | 14.82 | 20.22 |
| ViT-B/16 (SWAG-LIN) | 3.57 | 2.56 | 2.75 | 0.64 | 8.28 | 10.42 | 16.29 | 20.53 |
| ViT-L/16 | 3.08 | 2.29 | 2.47 | 0.70 | 6.75 | 8.66 | 11.39 | 14.22 |
| ViT-L/16 (SWAG-LIN) | 3.22 | 2.30 | 2.51 | 0.62 | 7.38 | 9.45 | 14.85 | 22.42 |
| ViT-H/14 (SWAG-LIN) | 3.16 | 2.28 | 2.48 | 0.66 | 7.13 | 9.05 | 13.48 | 18.63 |
| DINOv2 ViT-S/14 | 4.03 | 2.67 | 3.00 | 0.72 | 9.60 | 11.98 | 18.27 | 27.44 |
| DINOv2 ViT-B/14 | 3.92 | 2.58 | 2.84 | 0.78 | 9.28 | 12.29 | 19.41 | 26.05 |
| DINOv2 ViT-L/14 | 4.22 | 2.78 | 3.04 | 0.81 | 10.16 | 13.19 | 18.93 | 28.84 |
| DINOv2 ViT-g/14 | 4.01 | 2.60 | 2.92 | 0.74 | 9.64 | 13.20 | 21.04 | 25.13 |
| DINOv2 ViT-S/14 ColSelBoost | 4.12 | 2.83 | 3.15 | 0.77 | 9.72 | 12.06 | 18.30 | 24.43 |
| DINOv2 ViT-B/14 ColSelBoost | 3.95 | 2.71 | 2.98 | 0.83 | 9.21 | 11.81 | 19.06 | 24.61 |
| DINOv2 ViT-L/14 ColSelBoost | 4.27 | 2.81 | 3.12 | 0.79 | 10.26 | 12.89 | 19.05 | 29.68 |
| DINOv2 ViT-g/14 ColSelBoost | 3.99 | 2.59 | 2.91 | 0.74 | 9.65 | 12.71 | 19.47 | 25.55 |

**Table 9**
Color selectivity statistics for the neurons belonging to the ResNet-18, ViT, and DINOv2 features: ratio of non color selective, low color selective and color selective neurons (left); categorization of color selective neurons on the basis of the number of color ranges to which they are selective (right).

| Selectivity | No. of neur. | Non color select. | Low color select. | Color selective | Pan-chrom. | Single color range | Double color range | Triple(+) color range |
|---|---|---|---|---|---|---|---|---|
| ResNet-18 | 512 | 179 (35.0%) | 326 (63.7%) | 7 (1.4%) | 0 (0.0%) | 7 (100.0%) | 0 (0.0%) | 0 (0.0%) |
| ViT-B | 768 | 249 (32.4%) | 400 (52.1%) | 119 (15.5%) | 1 (0.8%) | 86 (72.3%) | 32 (26.9%) | 0 (0.0%) |
| ViT-B (SW.-L.) | 768 | 182 (23.7%) | 370 (48.2%) | 216 (28.1%) | 0 (0.0%) | 108 (50.0%) | 101 (46.8%) | 7 (3.2%) |
| ViT-L | 1024 | 321 (31.3%) | 496 (48.4%) | 207 (20.2%) | 0 (0.0%) | 96 (46.4%) | 107 (51.7%) | 4 (1.9%) |
| ViT-L (SW.-L.) | 1024 | 243 (23.7%) | 473 (46.2%) | 308 (30.1%) | 0 (0.0%) | 127 (41.2%) | 156 (50.6%) | 25 (8.1%) |
| ViT-H (SW.-L.) | 1280 | 262 (20.5%) | 540 (42.2%) | 478 (37.3%) | 0 (0.0%) | 161 (33.7%) | 280 (58.6%) | 37 (7.7%) |
| DINOv2-S | 384 | 215 (56.0%) | 135 (35.2%) | 34 (8.9%) | 1 (2.9%) | 28 (82.4%) | 5 (14.7%) | 0 (0.0%) |
| DINOv2-B | 768 | 331 (43.1%) | 359 (46.7%) | 78 (10.2%) | 0 (0.0%) | 67 (85.9%) | 11 (14.1%) | 0 (0.0%) |
| DINOv2-L | 1024 | 406 (39.6%) | 494 (48.2%) | 124 (12.1%) | 1 (0.8%) | 93 (75.0%) | 30 (24.2%) | 0 (0.0%) |
| DINOv2-g | 1536 | 734 (47.8%) | 664 (43.2%) | 138 (9.0%) | 0 (0.0%) | 96 (69.6%) | 42 (30.4%) | 0 (0.0%) |

## 4. Color selectivity

In this section we want to understand how the different models encode the color information within the extracted features. To this end, we analyze the color selectivity of the different neurons contained in the layer used for feature extraction, as the features are essentially constituted by their activation values. We extend the method in [45, 46], since they focused only on non-negative convolutional neurons activation. The color selectivity index of a neuron $i$ is here estimated as the variation between its activation to color images with respect to its activation to their corresponding gray-level images. Starting from the set-1 of the Hue circle MNIST dataset, for each image we also consider its original gray-level version. Then, for each neuron we sort the activations on the color images in the dataset in descending order, and select the $N = 1000$ top images. Then we take the activations of the current neuron on the gray-level versions of the color images that produced the selected highest activations. The color selectivity index $\alpha_i$ of the neuron $i$ is computed as follows:

$$\alpha_i = \begin{cases} \infty & \text{if } \sum_{j\in J} w'_{i,j} > 0 \wedge \sum_{j\in J} w_{i,j} < 0 \\ \sum_{j\in J} w'_{i,j} / \sum_{j\in J} w_{i,j} & \text{otherwise} \end{cases} \quad (2)$$

where $J$ is the set of indexes of the color images that produced the $N$ highest activations for the neuron $i$, $w_{i,j}$ is the activation of the neuron $i$ to the $j$th color image, and $w'_{i,j}$ is the activation of the neuron $i$ to the $j$th gray-level image that corresponds to the $j$th color image.

On the basis of the selectivity index, individual neurons are classified as non color selective (with $\alpha_i > 0.90$), low color selective

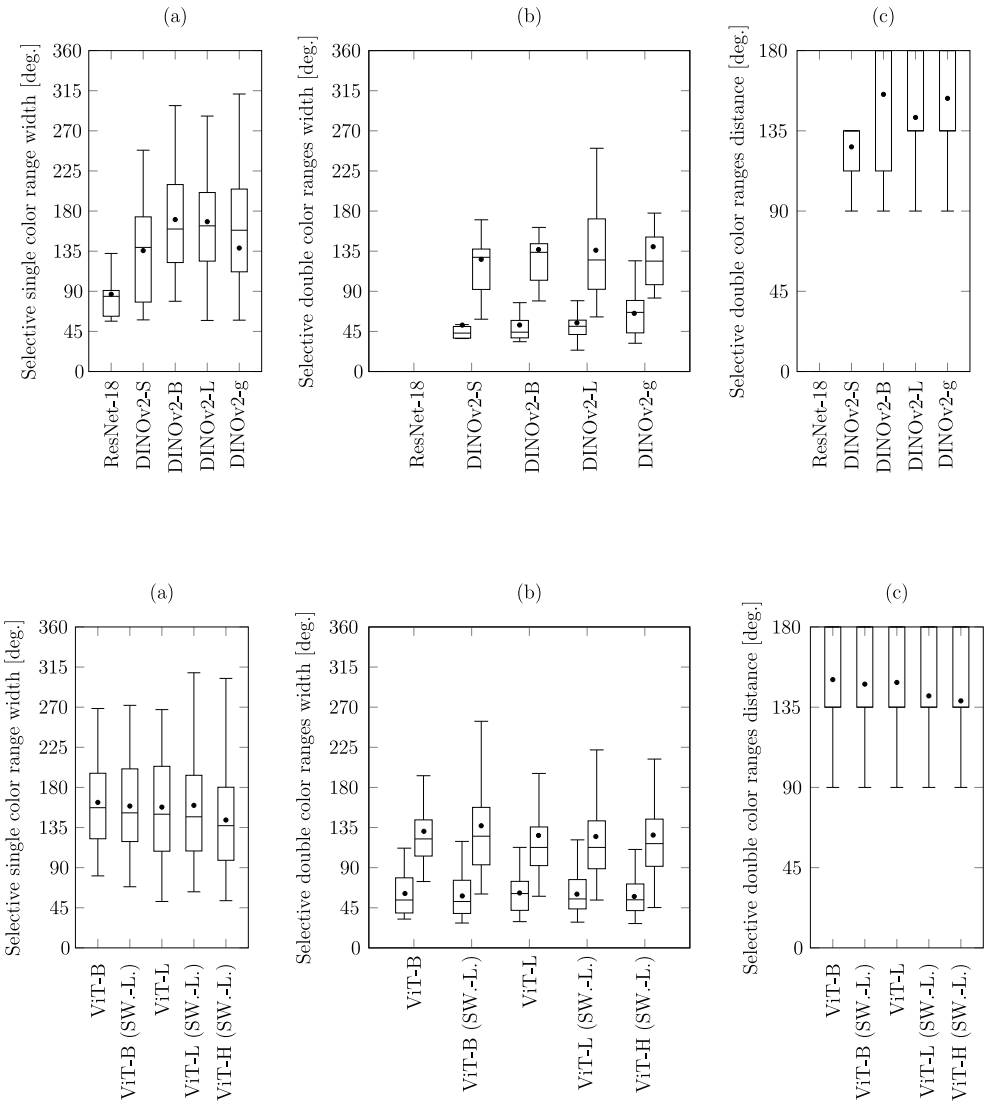


**Fig. 11.** Color selectivity properties for the ResNet-18, ViT, and DINOv2 features: (a) width of the single color range; (b) widths of the double color ranges; (c) angular distance between double color ranges.

(with $0.25 \leq \alpha_i \leq 0.90$), and color selective (with $\alpha_i < 0.25$). The results of this classification are reported in Table 9. From the values reported, it is possible to notice how ViT features have the highest ratio of color selective neurons (about 26.24% on average), followed by DINOv2 (about 10.05% on average), followed by ResNet-18. This clearly correlates with the performance obtained by each feature in the color vision task investigated in the previous section.

A further classification of color selective neurons into panchromatic (i.e., flat), single color range, double color range, and triple or more color ranges, depending on the number of color intervals to which they are selective, shows that ResNet-18 features only contain neurons selective to single color range. DINOv2 features mostly contain neurons selective to single color range, and a small ratio (about 20.9% on average) of neurons selective to double color ranges. ViT features instead contain about the same number of neurons selective to single and double color range, with a small ratio (about 4.2% on average) of neurons selective to three or more color ranges. For both DINOv2 and ViT features we observe how the number of neurons selective to double color ranges increases with model size.

As a further analysis, we want to understand some properties of the selective color ranges. In Fig. 11(a) we plot the width of the color interval that activates the most the neurons that are selective to single color range. We can observe that ResNet-18 neurons are selective for a an interval that is on average 90° wide. DINOv2 and ViT neurons instead are selective for a larger range, with an average width between 135° and 180°. The same analysis is also carried out for the double range color selective neurons. The widths of the intervals are plotted in Fig. 11(b), where it can be noticed that both DINOv2 and ViT neurons tends to have selective color intervals having two different widths: the first one with an average width of about 45°, and the second one with an average width of about 135°. Having two different intervals it is also interesting to understand the distance between them: the distance between interval peaks is plotted in Fig. 11(c). From the plots we can notice a high degree of color opponency among the color selective intervals, with an average distance larger than 135° for all models except for DINOv2-S, for which the distance is lower.

### 4.1. Color selectivity boost

In this section we want to find a general way to improve the performance of DINOv2 features on color vision tasks without any further training (e.g., [47]) or adaptation (e.g., [48]) to the specific

**Table 10**
Summary performance of ViT and DINOv2 features on the different color vision tasks considered, subdivided for the different dataset. For each column, the best performance is reported in bold and the second best is underlined. The last column reports the average rank of each feature on the different datasets.

| Method | Obj. color classif. | | | Localized attributes color classification | | | | | Color object segmentation | | | | Col. regr. | Avg rank |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Acc. ↑ | Acc. ↑ | Acc. ↑ | F. Acc. ↑ | B. Acc. ↑ | E. Acc. ↑ | H. Acc. ↑ | S. Acc. ↑ | BER ↓ | Prec. ↑ | Rec. ↑ | F1 ↑ | Ang. Err. ↓ | ↓ |
| ViT | **77.04** | **83.58** | 51.32 | 90.08 | 96.60 | 79.00 | **93.92** | **66.42** | 16.28 | 51.38 | 58.97 | 54.81 | **2.43** | 1.92 |
| DINOv2 | 45.38 | 75.43 | 42.68 | 79.43 | 92.18 | 78.23 | 78.28 | 38.10 | **7.93** | 70.74 | **75.37** | **72.98** | 2.66 | 2.38 |
| DINOv2 ColSelBoost | 58.60 | 82.20 | **51.48** | **91.33** | **97.20** | **80.13** | 86.85 | 53.83 | 8.35 | **70.78** | 74.39 | 72.53 | 2.74 | **1.69** |

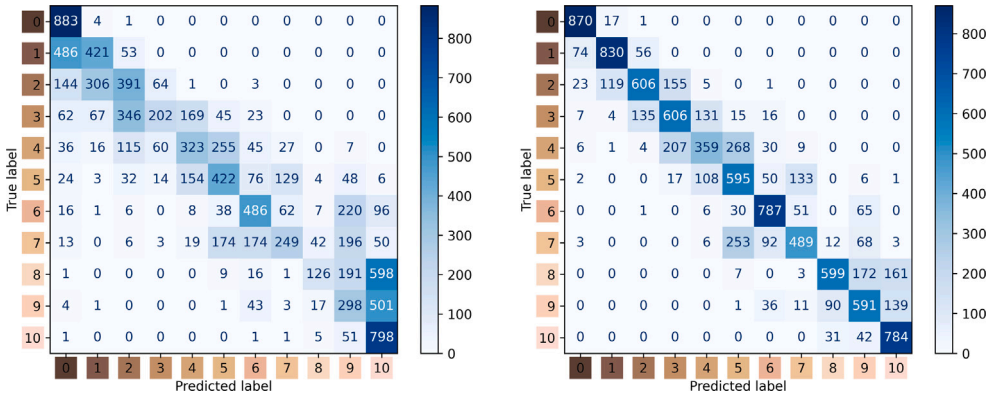


**Fig. 12.** Confusion matrices for skin color recognition on the `Google Cartoon Set`: DINOv2-g features (left) and DINOv2-g features with color selectivity boost (right). For each class we report the class label and its corresponding average skin color.

dataset. The idea investigated is to exploit the color sensitivity index computed in the previous section by changing in a non-linear way the contribution of color selective and non color selective neurons. Let $f = [f_1, \ldots, f_d]$ be the $d-$dimensional feature vector extracted from DINOv2; let $\bar{S} = \{\bar{s}_1, \ldots, \bar{s}_t\}$, with $t \leq d$, be the list of indexes of non color selective neurons within $f$. We then compute a new feature vector $F = [F_1, \ldots, F_d]$ where each element $F_k$ is computed as:

$$F_k = \begin{cases} sgn(f_k) \cdot |f_k|^{\alpha} & \text{if } k \notin \bar{S} \\ sgn(f_k) \cdot |f_k|^{\beta} & \text{if } k \in \bar{S} \end{cases} \tag{3}$$

The best configuration for $\alpha$ and $\beta$ is found by grid search in the solution space $(\alpha, \beta) \in \{0.5, 1, 2\}^2$ selecting the parameters yielding the highest average accuracy on the synthetic colored MNIST datasets introduced in Section 3.1, resulting in the optimal configuration $\alpha = 1$, $\beta = 0.5$. Being the input $f$ normalized to unitary length, the non-linear function family considered (i.e., power functions) assures no change for the extremal points (i.e., $\{-1, 0, 1\}$), includes the identity transformation ($\alpha = 1$, $\beta = 1$), and offers the possibility to be tuned for a specific dataset.

The performance reached by this color boosted feature vector are reported in the last section of Tables 4–8 and denoted as ColSelBoost. To help the evaluation of the impact of the color selectivity boost, a summary of the performance is reported in Table 10. We can notice that the proposed transformation is able to boost the performance of DINOv2 features in the object color classification and attribute color classification by 8.7% and 8.6% on average respectively. The most significant improvement is observed in the skin color recognition task, highlighting the transformation's effectiveness in distinguishing very similar color shades. The confusion matrices for DINOv2-g features and DINOv2-g features with color selective boost are reported in Fig. 12, where we can notice that the use of our transformation is able to reduce the number of off-by-one errors by about 36%. Furthermore, we can also notice that in several color classification and attribute color classification tasks the proposed transformation allows DINOv2 features to outperform ViT features. In the color object segmentation tasks instead, where DINOv2 features already outperform ViT ones, the proposed transformation does not give any boost in performance. Similarly, no improvement is observed in the color regression task, which requires specific domain training for optimal performance. Overall, averaging the rank reached by the different features on each dataset, we observe that the use of DINOv2 features with the proposed transformation is able to outperform supervised ViT features. In general, the use of the transformation introduced demonstrates the potential to effectively enhance DINOv2 capabilities in fine-grained classification tasks without retraining. This aligns with the broader observation that supervised ViT models excel in fine-grained, image-level tasks like classification and regression due to their higher proportion of color-selective neurons, while self-supervised DINOv2 models are more robust for general-purpose and pixel-level tasks like segmentation.

## 5. Conclusions

Computer vision foundation models have been demonstrated to produce visual features that even without any further fine-tuning work well in any computer vision task. In the literature this has been shown both on image-level and pixel-level tasks, while an assessment on vision tasks where color is a crucial information is missing. In this paper we assess the performance of DINOv2 features on different classes of color vision problems, from object color classification to object color segmentation.

The experimental results show that in image-level tasks, supervised ViT features are able to outperform self-supervised DINOv2 features. In order to understand the reason behind this, we performed a color selectivity analysis of both features, showing that ViT features tends to have a larger portion of color selective neurons, and among these a larger portion of neurons color selective for double, opponent, color ranges. Starting from this observation, we proposed a training-free transformation of DINOv2 features able to boost their color selectivity: with this transformation DINOv2 features were able to consistently reduce the gap in performance with ViT features, or even outperform them in some tasks. Concerning the pixel-level tasks considered, DINOv2 features are already able to outperform ViT features. The

main limitation of the proposed approach lies in the variability of the transformation's effectiveness across datasets with differing color distributions, which may require dataset-specific training.

As future works, we plan to extend the assessment to other foundation models on a larger set of color vision tasks, to further investigate the family of transformations able to provide a boost in color selectivity, and to explore the possibility of jointly training the parameters of the color selectivity boost transformation and the linear probe weights for each dataset.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

Data will be made available on request.

## References


[1] C. Zhou, Q. Li, C. Li, J. Yu, Y. Liu, G. Wang, K. Zhang, C. Ji, Q. Yan, L. He, et al., A comprehensive survey on pretrained foundation models: A history from bert to chatgpt, Int. J. Mach. Learn. Cybern. (2024) 1–65.

[2] M. Awais, M. Naseer, S. Khan, R.M. Anwer, H. Cholakkal, M. Shah, M.-H. Yang, F.S. Khan, Foundational models defining a new era in vision: A survey and outlook, 2023, arXiv preprint arXiv:2307.13721.

[3] T. Brown, B. Mann, N. Ryder, M. Subbiah, J.D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, Adv. Neural Inf. Process. Syst. 33 (2020) 1877–1901.

[4] J. Kaplan, S. McCandlish, T. Henighan, T.B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, D. Amodei, Scaling laws for neural language models, 2020, arXiv preprint arXiv:2001.08361.

[5] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H.W. Chung, C. Sutton, S. Gehrmann, et al., Palm: Scaling language modeling with pathways, J. Mach. Learn. Res. 24 (240) (2023) 1–113.

[6] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al., Llama: Open and efficient foundation language models, 2023, arXiv preprint arXiv:2302.13971.

[7] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in: International Conference on Machine Learning, PMLR, 2021, pp. 8748–8763.

[8] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, A. Joulin, Emerging properties in self-supervised vision transformers, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 9650–9660.

[9] R. Girdhar, A. El-Nouby, Z. Liu, M. Singh, K.V. Alwala, A. Joulin, I. Misra, Imagebind: One embedding space to bind them all, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 15180–15190.

[10] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A.C. Berg, W.-Y. Lo, et al., Segment anything, 2023, arXiv preprint arXiv:2304.02643.

[11] M. Oquab, T. Darcet, T. Moutakanni, H.V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. HAZIZA, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, P. Bojanowski, DINOv2: Learning robust visual features without supervision, Trans. Mach. Learn. Res. (2024).

[12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, 2020, arXiv preprint arXiv:2010.11929.

[13] J.P. Huix, A.R. Ganeshan, J.F. Haslum, M. Söderberg, C. Matsoukas, K. Smith, Are natural domain foundation models useful for medical image classification? in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2024, pp. 7634–7643.

[14] A. Shtedritski, C. Rupprecht, A. Vedaldi, What does clip know about a red circle? Visual prompt engineering for vlms, 2023, arXiv preprint arXiv:2304.06712.

[15] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2009, pp. 248–255.

[16] M. Singh, L. Gustafson, A. Adcock, V. de Freitas Reis, B. Gedik, R.P. Kosaraju, D. Mahajan, R. Girshick, P. Dollár, L. Van Der Maaten, Revisiting weakly supervised pre-training of visual perception models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 804–814.

[17] K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks, in: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, the Netherlands, October 11–14, 2016, Proceedings, Part IV 14, Springer, 2016, pp. 630–645.

[18] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.

[19] J. von Kries, Theoretische studien über die umstimmung des sehorgans, Festschr. Albrecht-Ludwigs- Univ. (1902) 145–158.

[20] CIE Standard, et al., 014-4/E: 2007, colorimetry-Part 4: CIE 1976 L*a*b* colour spaces, in: Commission Internationale de l'Eclairage, Wien, Austria, 2007.

[21] K. Panetta, L. Kezebou, V. Oludare, J. Intriligator, S. Agaian, Artificial intelligence for text-based vehicle search, recognition, and continuous localization in traffic videos, AI 2 (4) (2021) 684–704.

[22] P. Chen, X. Bai, W. Liu, Vehicle color recognition on urban road by feature context, IEEE Trans. Intell. Transp. Syst. 15 (5) (2014) 2340–2346.

[23] J. Wang, X. Li, J. Yang, Stacked conditional generative adversarial networks for jointly learning shadow detection and shadow removal, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1788–1797.

[24] V. Nguyen, T.F. Yago Vicente, M. Zhao, M. Hoai, D. Samaras, Shadow detection with conditional generative adversarial networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 4510–4518.

[25] T.F.Y. Vicente, L. Hou, C.-P. Yu, M. Hoai, D. Samaras, Large-scale training of shadow detectors with noisily-annotated shadow examples, in: Computer Vision–ECCV 2016: 14th European Conference, Proceedings, Part VI 14, 2016, pp. 816–832.

[26] K. Hashemifard, F. Florez-Revuelta, From garment to skin: The visuaal skin segmentation dataset, in: International Conference on Image Analysis and Processing, Springer, 2022, pp. 59–70.

[27] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440.

[28] V. Badrinarayanan, A. Kendall, R. Cipolla, Segnet: A deep convolutional encoder-decoder architecture for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (12) (2017) 2481–2495.

[29] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conf., 2015, pp. 234–241.

[30] X. Feng, X. Gao, L. Luo, Hlnet: a unified framework for real-time segmentation and facial skin tones evaluation, Symmetry 12 (11) (2020) 1812.

[31] M.K. Hasan, L. Dahal, P.N. Samarakoon, F.I. Tushar, R. Martí, DSNet: Automatic dermoscopic skin lesion segmentation, Comput. Biol. Med. 120 (2020) 103738.

[32] S. Fu, M. Hamilton, L. Brandt, A. Feldman, Z. Zhang, W.T. Freeman, Featup: A model-agnostic framework for features at any resolution, 2024, arXiv preprint arXiv:2403.10516.

[33] D. Liu, J. Liang, T. Geng, A. Loui, T. Zhou, Tripartite feature en-hanced pyramid network for dense prediction, IEEE Trans. Image Process. 32 (2023) 2678–2692.

[34] P.V. Gehler, C. Rother, A. Blake, T. Minka, T. Sharp, Bayesian color constancy revisited, in: 2008 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2008, pp. 1–8.

[35] D. Liu, Y. Cui, L. Yan, C. Mousas, B. Yang, Y. Chen, Densernet: Weakly supervised visual localization using multi-scale feature aggregation, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, 2021, pp. 6101–6109, 7.

[36] J.-M. Geusebroek, R. Van den Boomgaard, A.W.M. Smeulders, H. Geerts, Color invariance, IEEE Trans. Pattern Anal. Mach. Intell. 23 (12) (2001) 1338–1350.

[37] S.D. Hordley, G.D. Finlayson, Reevaluation of color constancy algorithm performance, J. Opt. Soc. Amer. A 23 (5) (2006) 1008–1020.

[38] J. Van De Weijer, T. Gevers, A. Gijsenij, Edge-based color constancy, IEEE Trans. Image Process. 16 (9) (2007) 2207–2214.

[39] J.T. Barron, Y.-T. Tsai, Fast fourier color constancy, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 886–894.

[40] Y. Hu, B. Wang, S. Lin, Fc4: Fully convolutional color constancy with confidence-weighted pooling, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4085–4094.

[41] H. Gong, Convolutional mean: A simple convolutional neural network for illuminant estimation, in: British Machine Vision Conference, 2019.

[42] S. Bianco, C. Cusano, Quasi-unsupervised color constancy, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 12212–12221.

[43] S. Bianco, C. Cusano, R. Schettini, Color constancy using CNNs, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2015, pp. 81–89.

[44] Z. Lou, T. Gevers, N. Hu, M.P. Lucassen, et al., Color constancy by deep learning, in: BMVC, 2015, 76–1.

[45] I. Rafegas, M. Vanrell, Color representation in CNNs: parallelisms with biological vision, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2017, pp. 2697–2705.
[46] I. Rafegas, M. Vanrell, Color encoding in biologically-inspired convolutional neural networks, Vis. Res. 151 (2018) 7–17.
[47] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, Q. He, A comprehensive survey on transfer learning, Proc. IEEE 109 (1) (2020) 43–76.
[48] E.J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, Lora: Low-rank adaptation of large language models, 2021, arXiv preprint arXiv: 2106.09685.

**Simone Bianco** is Associate professor of Computer Science at the University of Milano–Bicocca, holder of the Italian National Academic Qualification as Full Professor of Computer Engineering (09/H1) and Computer Science (01/B1). He is on Stanford University's World Ranking Scientists List for his achievements in Artificial Intelligence and Image Processing. His teaching and research interests include computer vision, artificial intelligence, machine learning, optimization algorithms applied in multimodal and multimedia applications. He is R&D Manager of the University of Milano Bicocca spin off "Imaging and Vision Solutions", and member of ELLIS (European Laboratory for Learning and Intelligent Systems).