Pattern Recognition Letters 128 (2019) 38–44

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

# Visual representation decoding from human brain activity using machine learning: A baseline study

Angeliki Papadimitriou*, Nikolaos Passalis, Anastasios Tefas

*Artificial Intelligence and Information Analysis Laboratory, Aristotle University of Thessaloniki, Thessaloniki, 541 24, Greece*

ARTICLE INFO

*Article history:*
Received 10 September 2018
Revised 24 May 2019
Accepted 6 August 2019
Available online 8 August 2019

*MSC:*
62M45
82C32
92B20
92C55

*Keywords:*
Neural decoding
Machine learning
Deep visual representations

ABSTRACT

Visual representation decoding refers to the task of deciphering what a subject is seeing or visualizing by observing the brain state via neuroimaging. In recent years, there is an increasing interest towards tackling the aforementioned task through the use of machine learning approaches. This study provides an extensive evaluation that will serve as a baseline for visual representation decoding, by exploring a wide range of model configurations, feature representations and evaluation setups. In this way, this work lays the groundwork for developing more sophisticated and accurate decoding pipelines. The evaluation results suggest that neural networks provide, on average, the best performance, while choosing the most appropriate similarity metric for the class decoding process depends mostly on the task at hand. Finally, this work may also assist domain experts to gain high-level insights about the brain's function, through several interesting observations, e.g., our findings hint brain regions that are dominant for specific tasks and back up related claims about potential correspondence of the cortical hierarchy with deep visual representations.



## 1. Introduction

Neural decoding is the process of cracking the code of the brain encoding mechanism by observing the brain activity of subjects executing a specific task. Brain activity can be recorded via functional neuroimaging methods that are capable of capturing the signal changes over time. Such methods are Electroencephalography (EEG) [1–4] and functional Magnetic Resonance Imaging (fMRI) [5–7] and and have provided data for several neural decoding endeavors in the literature, regarding tasks ranging from walking [8,9] and dancing [10] to sleep [11,12].

While neural decoding is established as the general task that aims to understand what the recorded brain activity represents in terms of stimulus or behavior, visual representation decoding is defined as a special case of neural decoding. This time, the signal to be decoded is the product of a *visual stimulus* that elicited the observed brain activity. This stimulus is encoded as an internal neural representation by the subject. The goal is to recover this representation by partially observing the brain state using a neuroimaging method, allowing for classifying or even reconstructing the visual stimulus that evoked the observed activity. The term *generic decoding* refers to the additional property that allows for decoding representations of images that belong to categories for which the model has not been trained. Hence, it is desirable to design a model that not only learns the mappings of fMRI data to images, but can also competently generalize to recognize an arbitrary stimulus.

To this day, the way the brain encodes and organizes visual information remains a partially understood -yet immensely complex- process and an area of active research in neuroscience [13]. The internal representations of the human brain are not known and, therefore, it is impossible to directly train a decoding model that translates brain activity signals to visual content. To overcome this limitation, an intermediate step is necessary to project both the visual content (images) and the brain activity data to a common representation space in order to successfully decode the neural activity. The work by Horikawa and Kamitani [14] hinted that the features extracted from various levels of a deep Convolutional Neural Network (CNN) are tightly correlated with the brain activity observed from various brain regions. Hence, CNN features can serve as the required (and appropriately regularized) intermediate neural representation that will assist a machine learning model [15] to recognize patterns in the observed brain signals and ultimately predict what a subject actually sees or imagines [16].

The main contribution of this paper is to provide a solid baseline for the task of decoding generic visual representations from brain activity. In recent years, there is an increasing interest in visual representation decoding with several methodologies being

* Corresponding author.
*E-mail address:* akpapadim@csd.auth.gr (A. Papadimitriou).

https://doi.org/10.1016/j.patrec.2019.08.007
0167-8655/

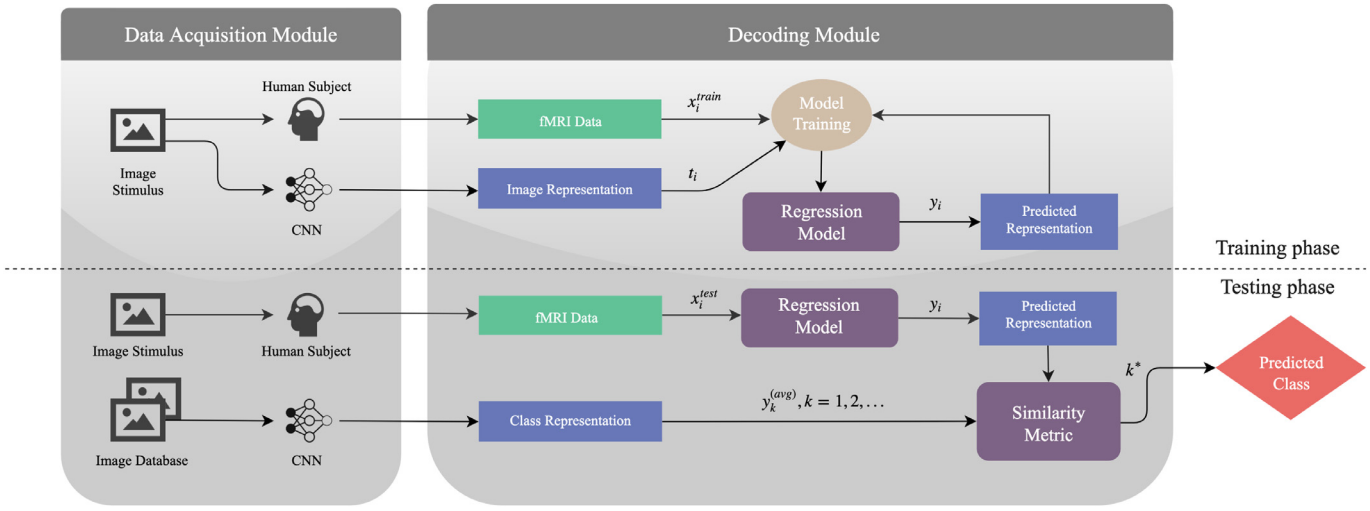


**Fig. 1.** The pipeline for visual representation decoding from human brain activity data. fMRI data represents recordings from different brain regions (V1,V2,V3...). Image/Class Representations are the feature vectors produced by different network layers (CNN1, CNN2, CNN3...). Similarity metric (Pearson, euclidean, cosine) measures how similar the model's prediction is to every image class in the database, and serves as the ultimate criterion to elect the top candidate class.

proposed [14,16–18]. However, to the best of our knowledge, there exists no comprehensive assessment on the effect of different combinations of models, metrics and hyper-parameters on the decoding performance. For instance, the work of Horikawa and Kamitani [14] evaluates only one regression model and employs only one metric to assess the performance. In this paper, we extensively evaluate different design choices for several parts of the decoding pipeline proposed by Horikawa and Kamitani [14], such as target features, regression models and similarity metrics. This experimentation with different models not only improves the decoding accuracy but also indicates that the two visual tasks point out to different regression models. Another issue is that [14] provides only graphical results that present a comparative summary of the model's decoding performance. Thus, there is no accurate way to compare other newly proposed models or decoding pipelines with the existing findings. Hence, the need for an established baseline for the task of visual representation decoding becomes apparent. In this work, we provide the complete numerical results as supplementary material, to facilitate comparison between different approaches and serve as a reference point to develop more sophisticated and accurate decoding methodologies. The importance of the reported results is further validated by statistical tests. The code used to perform the experiments is available on-line at https://github.com/angpapadi/Visual-Representation-Decoding-from-Brain-Activity allowing for easily reproducing the performed experiments and comparing newly proposed methods for the task.

The rest of this paper is structured as follows. The methodology applied in the present study, along with the machine learning models and evaluation setups are described in Section 2, while the used dataset, experimental protocols and extensive evaluation results are presented in Section 3. Finally, conclusions are drawn in Section 4.

## 2. Generic brain activity decoding

We follow the generic brain activity decoding pipeline proposed in [14]. The employed pipeline is summarized in Fig. 1. First, a subject performs a visual task, while his/her brain activity is monitored using fMRI. Also, a CNN is used to extract a feature representation from the corresponding images (as they are presented to the subject). These representations can serve as a proxy for decoding the human brain activity for specific visual tasks, i.e., recognizing the class of an image shown to or visualized by a subject. To decode the brain activity, a regression model is used to directly predict the representation of the stimulus image using as sole input the measured fMRI signals. Subsequently, the class of the stimulus is deducted by comparing the decoded representation to a set of prototype class representation vectors. Note that this regression-based pipeline can be effectively used to infer the category/class for images which were *never presented* to the subject and/or regression model during the training.

The decoding pipeline is formally defined as follows. Let $\mathbf{x}_i \in \mathbb{R}^N$ be a $N$-dimensional feature vector that is extracted from the measured fMRI signals when the $i$-th experiment is performed [14]. Also, the notation $\mathbf{t}_i \in \mathbb{R}^L$ is used to denote the representation extracted from a layer of a CNN, when the image used for the $i$-th experiment is fed to the network, while $L$ is the dimensionality of this representation. Apart from the image representations, that are directly extracted from the CNN, prototype class representation are compiled for each class. This allows for generic class decoding for classes that were never presented to the model/subject during the training. The prototype class representation for the $k$-th class is defined as:

$$\mathbf{y}_k^{(avg)} = \frac{1}{|\mathcal{R}_k|} \sum_{\mathbf{y} \in \mathcal{R}_k} \mathbf{y} \in \mathbb{R}^L, \tag{1}$$

where $\mathcal{R}_k$ is the set of CNN representations extracted from the images that belong to class $k$ and $|\mathcal{R}_k|$ is the size of this set. A machine learning model $f_{\mathbf{W}}(\mathbf{x})$ is then used to regress the image representation $\mathbf{t}_i$ using as input the corresponding brain activity $\mathbf{x}_i$. The notation $\mathbf{W}$ is used to indicate the parameters of the employed model. The output of this model is calculated as $\mathbf{y}_i = f_{\mathbf{W}}(\mathbf{x}_i) \in \mathbb{R}^L$ and can be used to infer the class of the corresponding image by measuring the similarity of $\mathbf{y}_i$ with each of the class representations $\mathbf{y}_k^{(avg)}$. Therefore, the predicted class $k^*$ of the object the subject sees or imagines is calculated as:

$$k^* = \arg\max_k S(\mathbf{y}_i, \mathbf{y}_k^{(avg)}), \tag{2}$$

where $S(\mathbf{y}_1, \mathbf{y}_2)$ is an appropriately defined similarity metric between two vectors $\mathbf{y}_1 \in \mathbb{R}^L$ and $\mathbf{y}_2 \in \mathbb{R}^L$. Note that (2) essentially describes a Nearest Centroid Classifier (NCC) [19]. The CNN representations $\mathbf{t}_i$ are only needed during the training process, as also shown in Fig. 1. Note that the image representations are not needed during the inference (test), since the class of the object

that a subject sees or imagines can be deduced using only the (precomputed) class representation vectors.

### 2.1. Regression models

As mentioned in Section 1, the objective of this baseline study is, inter alia, to assess the performance of various machine learning models for the task of decoding the brain representations to the target feature vectors. In the present study, four different regression models are evaluated. The employed models are introduced below:

1. **k-Nearest Neighbor Regression (kNN):** The $k$ nearest neighbors of a test sample are used in k-Nearest Neighbor Regression to infer the output representation by averaging the (known) target vectors of its neighbors in the train set [20]. To take into account the proximity of each training sample to the current sample, the contribution of each neighbor is weighted according to its similarity to the current input.
2. **Linear Regression (LR):** Linear Regression is a well-known and widely used regression model [21–23]. The output of LR is calculated as: $f_{\mathbf{W}}(\mathbf{x}) = \mathbf{W}_{lr}\mathbf{x} + \mathbf{b}_{lr}$ [24], where $\mathbf{W}_{lr} \in \mathbb{R}^{L \times N}$ is the matrix that contains the model's parameters and $\mathbf{b}_{lr} \in \mathbb{R}^{L}$ is a vector that contains the independent terms. The mean squared error between the target representation and the output of the model is used for training:

$$\mathcal{L}_{mse} = \frac{1}{2N} \sum_{i} ||f_{\mathbf{W}}(\mathbf{x}_i) - \mathbf{t}_i||_2^2, \tag{3}$$

   where the notation $||\mathbf{x}||_2$ is used to denote the $l^2$ norm of a vector $\mathbf{x}$ and $N$ is the number of samples used for training the regression model. The model can be also *regularized* to avoid overfitting phenomena. For example, in Ridge Regression (RR) [25], the $l^2$ norm of the regression parameters is used to this end.
3. **Kernel Regression (KR):** Kernel Regression is a powerful non-linear variant of LR. In KR the data are first projected into a higher-dimensional space, where they can be better separated [26,27], using the so-called *kernel trick*.
4. **Multilayer Perceptrons (MLP):** Multilayer Perceptrons are powerful models that can model complex non-linear relationships between the input data and their targets using multiple layers [28]. Note that MLPs are often prone to overfitting, especially when a small number of training samples is used, leading to the development of several regularization methods, such as Dropout [29] (also abbreviated as "drop." in this paper). The networks used in this paper are trained to minimize the squared error loss function given in (3) using the Adam optimizer [30].

### 2.2. Class prediction

The resulting regressed vectors are provided as input to a class decoder that selects the most probable image class. The original setup proposed by Horikawa and Kamitani [14] employs a similarity metric to compare the decoded feature vector to each of the class representations. The class, whose representation is most similar to the regressed vector, is the best candidate image class as described in (2). In this paper we thoroughly study the effect of different similarity metrics on the accuracy of the decoding. The following similarity metrics are considered:

1. **Euclidean similarity:** The Euclidean similarity is computed as the inverse of Euclidean distance:

$$S_{euclidean}(\mathbf{y}_1, \mathbf{y}_2) = \frac{1}{1 + ||\mathbf{y}_1 - \mathbf{y}_2||_2}. \tag{4}$$

2. **Cosine similarity:** The cosine similarity is defined as the angle between two vectors:

$$S_{cosine}(\mathbf{y}_1, \mathbf{y}_2) = \frac{\mathbf{y}_1^T \mathbf{y}_2}{||\mathbf{y}_1||_2 ||\mathbf{y}_2||_2}. \tag{5}$$

3. **Pearson similarity:** The Pearson similarity (correlation) between two vectors is computed as:

$$S_{pearson}(\mathbf{y}_1, \mathbf{y}_2) = \frac{(\mathbf{y}_1 - \mu_1)^T (\mathbf{y}_2 - \mu_2)}{||\mathbf{y}_1 - \mu_1||_2 ||\mathbf{y}_2 - \mu_2||_2}, \tag{6}$$

   where the notation $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ is used to denote the average of the values in vectors $\mathbf{y}_1$ and $\mathbf{y}_2$ respectively.

### 2.3. Evaluation protocol

Using the setup proposed in [14], the model's accuracy is calculated by determining the percentage of the candidate categories for which the similarity between the given category and a test sample is smaller than the similarity with its correct class.

## 3. Experimental evaluation

This section provides the experimental evaluation details. First, the used dataset and feature extraction methods are described in Section 3.1, while the different experimental setups are introduced in Section 3.2. Finally, the experimental results are presented and discussed in Section 3.3.

### 3.1. Data

In the present study, we use the publicly available dataset described by Horikawa and Kamitani [14]. It consists of fMRI data from 5 human subjects performing visual tasks. Note that we cannot train a model using the data of all 5 subjects, since the fMRI representations extracted from different subjects can vary widely and performing transfer learning from one subject's encoding to another's is not trivial [18]. Instead, a separate encoding model is trained and evaluated individually for every subject and the performance is averaged across all subjects in the final evaluation step. In total, for every subject, there exist 3450 brain activity samples, 1200 for training and 2250 for testing, each corresponding to a visual stimulus. The stimuli images are natural images from the ImageNet dataset [31]. Notably, the images that form the train and test sets belong to non-overlapping classes, i.e., ImageNet synsets, prohibiting the use of traditional classification algorithms for the task of neural decoding. In particular, the train set consists of 150 image classes and the test set of 50 image classes.

The data cover two distinct visual tasks (Fig. 2) performed by each subject: (a) The image presentation task, where images were directly shown to the subjects and (b) the imagery task, where the subjects were instructed to visualize images that correspond to a given word. The dataset contains a total of 1750 test samples per subject for the image presentation experiments and 500 test samples per subject for the imagery experiments. The interested reader is referred to [14] for more details regarding the data collection protocol and experimental setups.

Each data sample contains information from multiple brain regions specialized in visual processing, that are summarized in Fig. 3. Specifically, the data include voxel information from:

- The areas V1, V2, V3 and V4, that are responsible for detecting edges, color and contours,
- The lateral occipital complex (LOC), that is involved with shape and object processing,
- The fusiform face area (FFA), that is specialized in the encoding and recognition of faces, and

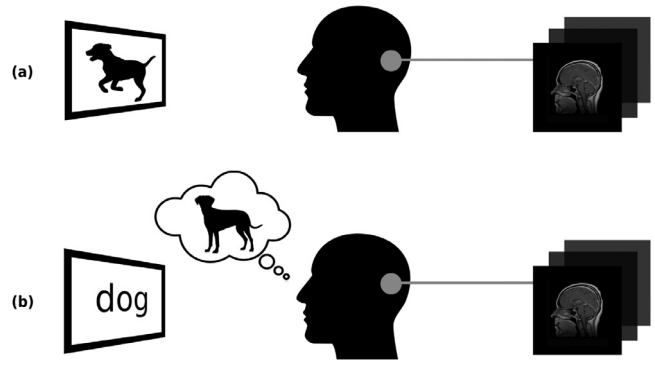


**Fig. 2.** Schematic view of the visual tasks performed by the human subjects for a given period of time while their brain activity is recorded via fMRI (a) In the image presentation task, the subject sees an image of a specific category (b) In the imagery task, the subject sees a word that corresponds to a specific image category and is instructed to visualize images that are relevant to the concept corresponding to the word he/she sees.

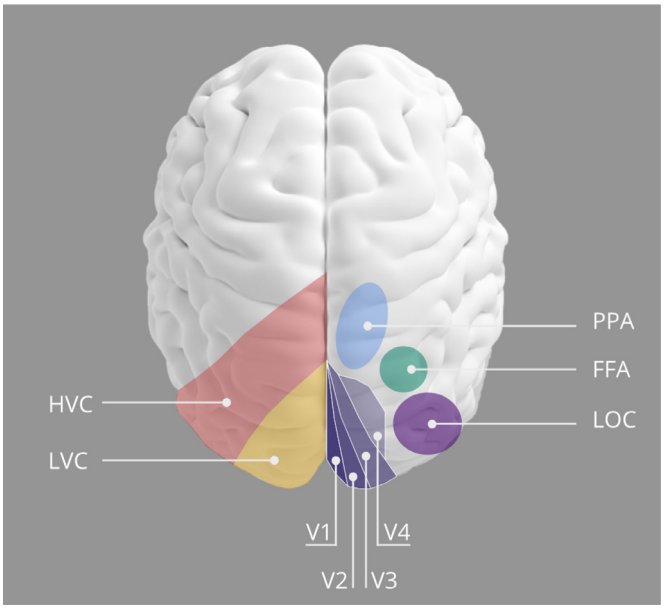


**Fig. 3.** Schematic view of the visual cortex with highlighted regions of interest.

- The parahippocampal place area (PPA), that is responsible for spatial information encoding.

Hence, three clusters of regions can be formed to provide a hierarchical understanding of the visual system: (a) The lower visual cortex (LVC) that includes voxels from areas V1-V3, (b) the higher visual cortex that comprises regions LOC, FFA, PPA and (c) the entire visual cortex (VC) that encompasses all the aforementioned sub-regions. Note that for areas V1, V2, V3, V4, LOC, FFA and PPA, all voxels lying near the border, are included in both the neighboring regions. Thus, there exists some overlap between those areas. All the extracted voxels from the corresponding brain regions were used in the conducted experiments, i.e., we did not select only the voxels with the highest correlation, as in [14]. This allows for retaining as much information as possible. Additional information on voxel information data is provided in the supplementary material.

The images used as stimuli for the image presentation task are fed to a CNN that performs feature extraction. The use of a CNN layer for this task is not arbitrary. As discussed in [14], the features extracted from various layers of a CNN are tightly associated with the activity of various parts of the human visual cortex. The resulting feature vectors serve as the targets of the regression model that learns the mapping from recorded brain activity to natural images. Hence, every image is represented by an $L$-dimensional feature vector. In addition, for every class of images, a class representation is computed, as described in (1).

The AlexNet, which is composed of 5 convolutional (CNN1-CNN5) and 3 fully connected layers (FC1-FC3), was used to extract the target representations [32]. The model was pre-trained on the ImageNet dataset [32], while 1000 activations were randomly sampled from each layer to compile the feature representation [14]. For all reported experiments in this study, we used the available feature vectors, provided by the authors of [14] and produced as described above, to allow for easily reproducing the conducted experiments. Furthermore, we carried out multiple additional experiments using both image representations and class representations as targets.

### 3.2. Experimental setups

The conducted experiments include extensive evaluation of:

- Deep target features from different layers,
- Multiple regression models,
- Various similarity metrics,
- Using image representations as the regressor's targets compared to the class representations.

Unless explicitly stated, all reported experimental results use the feature representations extracted from layer CNN5 and are evaluated using the Pearson metric. Both the input features $\mathbf{x}_i$ and the visual representations $\mathbf{t}_i$ were normalized using z-score normalization, i.e., normalized to have zero mean and unit variance. Note that this implies that the output of the regression model must be re-centered and re-scaled using the computed mean and standard deviations before performing the class decoding. The following hyper-parameters were used for all the conducted experiments. The 5 nearest neighbors were used for regressing the representations using the kNN model. The weight of the regularizer was set to 1 for the RR, while a 2nd degree polynomial kernel led to the best performance for the KR model (the weight of the regularizer was set to 0.005, while a constant value of 10 was added to the kernel). After experimenting with various MLP architectures, an MLP with one hidden layer was used. The hidden layer is composed of 300 neurons that use sigmoid activation functions. The dropout probability was set to 30% (applied to the input layer only). The MLP models were trained for 100 epochs using batch size 128 employing the Adam optimizer (learning rate was set to 0.001 and the default hyper-parameters of the optimizer were used [30]). The interested reader is referred to the supplementary material for a table summarizing the range of values tested for each parameter of the experimental pipeline.

### 3.3. Experimental results

The extensive evaluation of the decoding pipeline has yielded some interesting findings. The first set of experiments examines the effect of choosing different machine learning models for the regression tasks of image presentation and imagery. The same experiments were conducted using both the image-specific targets (Image Representation) and the class-averaged targets (Class Representation). A summary of the results is showcased in Fig. 4.

For the image presentation task, the best accuracy score is achieved by the MLP regressor for the VC brain region. The predominance of the MLP model becomes even more evident when using the class-averaged representations (Fig. 4(c)). Regions V1-V4 seem to perform increasingly better, a finding that reflects their inherent hierarchical structure in the brain itself. Another interesting finding is that for the FFA region, non-linear models outperform

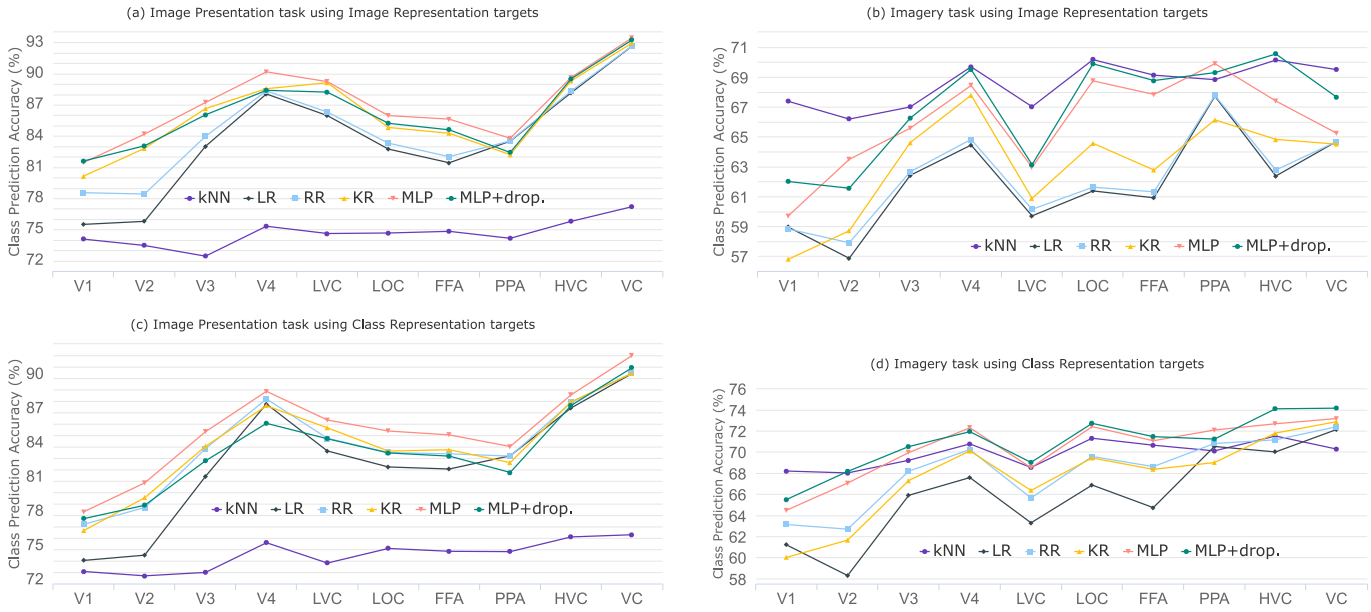


**Fig. 4.** Impact of various regression models on decoding accuracy.

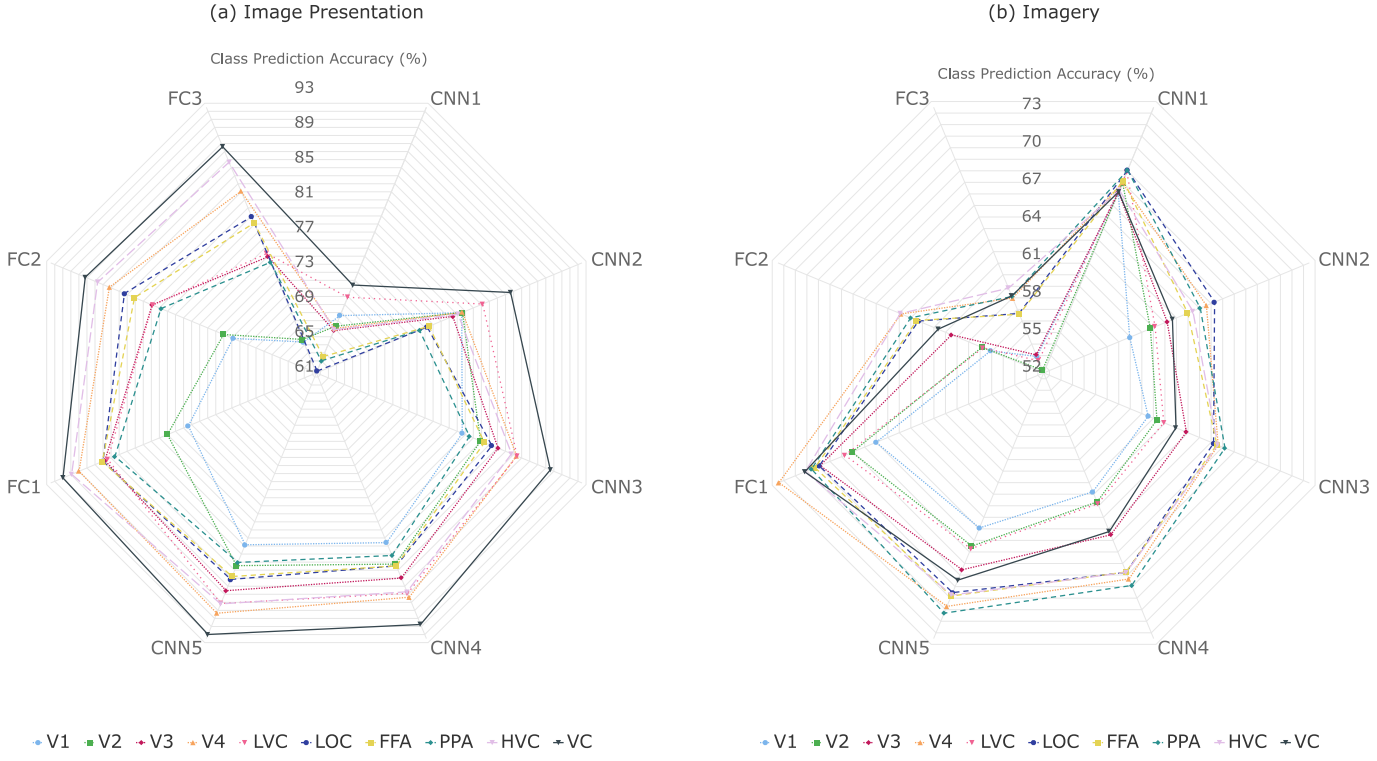


**Fig. 5.** Effect of using deep features from different layers for the (a) Image Presentation and (b) Imagery tasks.

the linear models, while for the PPA region the opposite is true. Moreover, HVC's performance exceeds the performance of each of its sub-regions. This finding suggests, that the voxels of the higher visual cortex areas provide complementary information, leading to much richer representations when processed as one structure. However, there is a drop in the accuracy score for all brain regions when the models were trained with the class representations. This may be explained by the fact that image representations preserve the features' diversity power, as opposed to the class representations that are averaged, and hence collapsed in the feature space.

Regarding the imagery task, this trend is reversed. Using class representations yields significantly better results than their image-specific counterparts (Fig. 4(d)). Indeed, the imagery task experimental protocol required that the subject visualized on cue as many images of a specific category as possible. Thus, the internal thought process of the subject might approximate the averaged class representation, explaining the better performance of these targets. This time, the top performing models are kNN and MLP+drop., both in the HVC region, for the image-specific and class-averaged targets respectively. In general, kNN seems to be

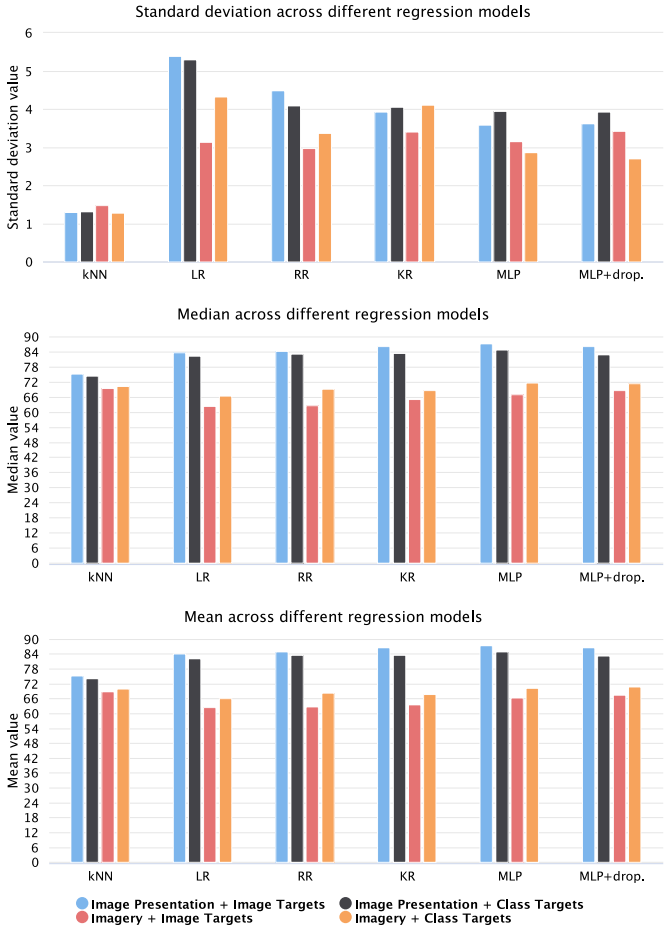


**Fig. 6.** Comparison of standard deviation, median and mean value across different regression models.

more robust to distribution shifts caused by feeding the brain activity features of the imagery task to the model, since its output is always a constrained linear combination of the training targets.

Overall, there exist differences between the two visual tasks. Notably, VC is the dominant region in the image presentation task whereas in the imagery task, HVC is better or comparable to VC performance. In general, machine learning algorithms tend to perform better when exposed to more available information relevant to the task. This might explain why VC, the region that includes the total of a subject's voxel information, outperforms all other regions in the image presentation task. This is also the case for the results reported for HVC in the imagery task, that are consistently better than each of its subregions. Respectively, feeding a machine learning model with information that is irrelevant to the task at hand can hinder its performance. This seems to be the case for VC in the imagery task, as its performance is worse or comparable to its HVC subregion. In other words, the additional voxels of the VC region in the imagery task may function as noise, making it difficult for the model to make accurate predictions. By definition, the imagery task is harder to decode because the subjects were not fixated on one visual image but rather had to visualize/imagine several images related to a concept (Fig. 2).

Furthermore, the accuracy achieved in the imagery task is critically lower than the image presentation task. Apart from the case we made about noise in the recorded imagery patterns that could in part explain this dramatic drop in performance, we speculate that the imagery task as a brain process might carry a greater intrinsic variability, as the high-level encoding circuitry is not as genetically hard-coded as the primary visual cortex and its adjacent

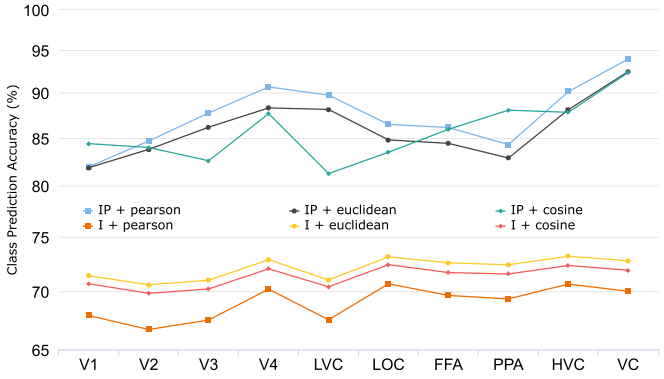


**Fig. 7.** Effect of different similarity metrics on the evaluation step for the Image Presentation (IP) and Imagery (I) tasks.

sub-regions. The latter structures, are present in all mammals [33], have been preserved for millions of years across species and, therefore, their function is far more homogeneous among humans compared to higher visual areas.

Our experimental results on the correspondence of certain brain regions with the levels of deep CNN representations, support the finding of [14] that suggests tight association between hierarchical visual areas and the complexity levels of visual features. Indeed, as illustrated in Fig. 5(a), for the image presentation task, features extracted from the convolutional layers tend to be better predicted by brain regions of the lower visual cortex. However, for the imagery task (Fig. 5(b)), higher cortex brain regions outperform the lower visual cortex sub-regions for all level representations, as was also the case in Fig. 4(b) and (d). If we assume that *the internal processes of the two tasks in the brain are similar*, this finding suggests that the lower visual cortex areas are not particularly involved during visual imagery tasks. This hypothesis is also supported by the fact that HVC alone is the top performing region for the imagery task, compared to the presentation task where VC, encompassing all sub-regions, scored the highest. If this hypothesis is false, then these results may hint that the analogy between CNNs and cortical hierarchy does not hold for tasks other than the image presentation. Nevertheless, the results clearly point to certain correlations between deep representations and brain regions, a property that can be further exploited to obtain better accuracy.

To confirm the validity of our experimental results, we conducted statistical tests on the findings reported by Figs. 4 and 5. The MLP models proved to be significantly better than the rest of the evaluated models. Similarly, feature representations from layers CNN4 and FC1 were deemed statistically significant. For a complete description of the Friedman tests we conducted, as well as figures of the results, please refer to the supplementary material.

For the sake of completeness, we provide charts (Fig. 6) reporting on the standard deviation, median and mean value of the accuracy scores of Fig. 4 obtained using each of the regression models. Similar charts were devised for the results reported in Fig. 5 and are available as supplementary material. In all cases, all distributions are approximately symmetrical as the median and mean values are almost equal.

The different similarity metrics (described in Section 2.3) were evaluated for the best performing model of both the image presentation and imagery tasks using the image representations as targets, i.e., for the MLP and kNN models respectively. As shown in Fig. 7, Pearson similarity yields the best results for the image presentation task, while Euclidean similarity considerably boosts the accuracy score on the imagery task.

For a complete report of the experimental results, that can be used for comparing new methods to the presented baselines and

generating accuracy plots, the reader is referred to the Supplementary Material.

## 4. Conclusions

An extensive evaluation of the decoding pipeline was performed in this paper, experimenting with various parts of the process, from the appropriate selection of the target features and regression models to the similarity metrics used in the class prediction step.

At this point, it is important to mention some limitations of CNNs, which were employed in the present work to encode the natural images used as stimuli. It has been shown that CNNs consistently fail to generalize when faced with adversarial perturbations or changes in the position, orientation and scale of the object of interest. By contrast, biological visual systems' performance is not affected by the aforementioned conditions. Even in tasks where CNN performance is comparable to humans, CNNs require a significantly larger number of examples to classify visual shapes. A recent study showed that humans not only remain unaffected in their capability to correctly categorize adversarial examples, but can also anticipate how CNNs will behave [34]. Moreover, studies conducted in [35] and [36] confirm that CNNs have limited abstracting capabilities as they are not invariant to translation, rotation and scale transformations in the input image. Several approaches try to tackle this problem, such as spatial transformer networks [37] that directly manipulate the feature map to account for affine transformations and capsule networks [38] that identify spatial relationships between different parts of an image. In this work, the dataset employed was already curated and all stimuli images were centered but "in the wild" datasets could really benefit from such methods.

Having said that, this experimental evaluation is expected to serve as a baseline for the task of visual representation decoding. This work provides empirical information about which models perform best for each visual task, how to determine which feature layer representation to use as the regressor's targets and what similarity metric most efficiently captures the complex relationships of the decoded representations. Interestingly, the reported results may also provide some insight regarding the brain's visual processing, as is hinted in Section 3.

## Declaration of Competing Interest

The authors declare no conflict of interest.

## Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.patrec.2019.08.007.

## References

[1] E. Niedermeyer, F.L. da Silva, Electroencephalography: Basic Principles, Clinical Applications, and Related Fields, Lippincott Williams & Wilkins, 2005.
[2] N. Arunkumar, K. Ramkumar, V. Venkatraman, E. Abdulhay, S.L. Fernandes, S. Kadry, S. Segal, Classification of focal and non focal eeg using entropies, Pattern Recognit. Lett. 94 (2017) 112–117.
[3] E. Pacola, V. Quandt, P. Liberalesso, S. Pichorim, F. Schneider, H. Gamba, A versatile eeg spike detector with multivariate matrix of features based on the linear discriminant analysis, combined wavelets, and descriptors, Pattern Recognit. Lett. 86 (2017) 31–37.
[4] R. Patel, K. Gireesan, S. Sengottuvel, Decoding non-linearity for effective extraction of the eye-blink artifact pattern from eeg recordings, Pattern Recognit. Lett. (2018).
[5] S.A. Huettel, A.W. Song, G. McCarthy, et al., Functional Magnetic Resonance Imaging, 1, Sinauer Associates, Sunderland, MA, 2004.
[6] C.O. Plumpton, Semi-supervised ensemble update strategies for on-line classification of FMRI data, Pattern Recognit. Lett. 37 (2014) 172–177.
[7] J. Amin, M. Sharif, M. Yasmin, S.L. Fernandes, A distinctive approach in brain tumor detection and classification using mri, Pattern Recognit. Lett. (2017).
[8] J.T. Gwin, K. Gramann, S. Makeig, D.P. Ferris, Removal of movement artifact from high-density eeg recorded during walking and running, J. Neurophysiol. 103 (6) (2010) 3526–3534.
[9] A. Presacco, R. Goodman, L. Forrester, J.L. Contreras-Vidal, Neural decoding of treadmill walking from noninvasive electroencephalographic signals, J. Neurophysiol. 106 (4) (2011) 1875–1887.
[10] J.G. Cruz-Garza, Z.R. Hernandez, S. Nepaul, K.K. Bradley, J.L. Contreras-Vidal, Neural decoding of expressive human movement from scalp electroencephalography (EEG), Front. Hum. Neurosci. 8 (2014) 188.
[11] E. Tagliazucchi, H. Laufs, Decoding wakefulness levels from typical fmri resting-state data reveals reliable drifts between wakefulness and sleep, Neuron 82 (3) (2014) 695–708.
[12] T. Horikawa, M. Tamaki, Y. Miyawaki, Y. Kamitani, Neural decoding of visual imagery during sleep, Science 340 (6132) (2013) 639–642.
[13] J. Winawer, N. Witthoft, Identification of the ventral occipital visual field maps in the human brain, F1000Research 6 (2017).
[14] T. Horikawa, Y. Kamitani, Generic decoding of seen and imagined objects using hierarchical visual features, Nat. Commun. 8 (2017) 15037.
[15] N.M. Nasrabadi, Pattern recognition and machine learning, J. Electron Imag. 16 (4) (2007) 049901.
[16] H. Wen, J. Shi, Y. Zhang, K.-H. Lu, J. Cao, Z. Liu, Neural encoding and decoding with deep learning for dynamic natural vision, Cerebral Cortex (2017) 1–25.
[17] Y. Güçlütürk, U. Güçlü, K. Seeliger, S. Bosch, R. van Lier, M.A. van Gerven, Reconstructing perceived faces from brain activations with deep adversarial neural decoding, in: Proceedings of the Advances in Neural Information Processing Systems, 2017, pp. 4246–4257.
[18] H. Wen, J. Shi, W. Chen, Z. Liu, Transferring and generalizing deep-learning-based neural encoding models across subjects, Neuroimage 176 (2018) 152–163.
[19] H. Schütze, C.D. Manning, P. Raghavan, Introduction to Information Retrieval, 39, Cambridge University Press, 2008.
[20] T. Hastie, R. Tibshirani, Discriminant adaptive nearest neighbor classification and regression, in: Proceedings of the Advances in Neural Information Processing Systems, 1996, pp. 409–415.
[21] M.A. Domingues, R.M. de Souza, F.J.A. Cysneiros, A robust method for linear regression of symbolic interval data, Pattern Recognit. Lett. 31 (13) (2010) 1991–1996.
[22] J. Cui, Q. Zhu, D. Wang, Z. Li, Learning robust latent representation for discriminative regression, Pattern Recognit. Lett. (2018).
[23] J. Yuan, D. Wang, R. Li, Image segmentation using local spectral histograms and linear regression, Pattern Recognit. Lett. 33 (5) (2012) 615–622.
[24] F. Mosteller, J.W. Tukey, Data analysis and regression: a second course in statistics., Addison-Wesley Ser. Behav. Sci. (1977).
[25] A.E. Hoerl, R.W. Kennard, Ridge regression: biased estimation for nonorthogonal problems, Technometrics 12 (1) (1970) 55–67.
[26] J. Shawe-Taylor, N. Cristianini, et al., Kernel Methods for Pattern Analysis, Cambridge University Press, 2004.
[27] J. Liu, F. Zhao, Y. Liu, Learning kernel parameters for kernel fisher discriminant analysis, Pattern Recognit. Lett. 34 (9) (2013) 1026–1031.
[28] S.S. Haykin, Neural Networks and Learning Machines, 3, Pearson Education Upper Saddle River, 2009.
[29] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (1) (2014) 1929–1958.
[30] D. Kingma, J. Ba, Adam: a method for stochastic optimization, in: Proceedings of the International Conference on Learning Representations, 2014.
[31] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: a large-scale hierarchical image database, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255.
[32] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Proceedings of the Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[33] L. Krubitzer, The magnificent compromise: cortical field evolution in mammals, Neuron 56 (2) (2007) 201–208.
[34] Z. Zhou, C. Firestone, Humans can decipher adversarial images, Nat. Commun. 10 (1) (2019) 1334.
[35] S. Stabinger, A. Rodríguez-Sánchez, J. Piater, 25 years of CNNs: can we compare to human abstraction capabilities? in: International Conference on Artificial Neural Networks, Springer, 2016, pp. 380–387.
[36] J. Kim, M. Ricci, T. Serre, Not-so-clevr: learning same–different relations strains feedforward neural networks, Interface Focus 8 (4) (2018) 20180011.
[37] M. Jaderberg, K. Simonyan, A. Zisserman, et al., Spatial transformer networks, in: Advances in neural information processing systems, 2015, pp. 2017–2025.
[38] S. Sabour, N. Frosst, G.E. Hinton, Dynamic routing between capsules, in: Advances in neural information processing systems, 2017, pp. 3856–3866.