Towards Disentangling the Roles of Vision & Language in Aesthetic Experience with Multimodal DNNs

Anonymous Author(s)

Abstract

When we experience a visual stimulus as beautiful, how much of that response is the product of ineffable perceptual computations we cannot readily describe versus semantic or conceptual knowledge we can easily translate into natural language? Disentangling perception from language in any experience (especially aesthetics) through behavior or neuroimaging is empirically laborious, and prone to debate over precise definitions of terms. In this work, we attempt to bypass these difficulties by using the learned representations of deep neural network models trained exclusively on vision, exclusively on language, or a hybrid combination of the two, to predict human ratings of beauty for a diverse set of naturalistic images by way of linear decoding. We first show that while the vast majority (∼75%) of explainable variance in human beauty ratings can be explained with unimodal vision models (e.g. SEER), multimodal models that learn via language alignment (e.g. CLIP) do show meaningful gains (∼10%) over their unimodal counterparts (even when controlling for dataset and architecture). We then show, however, that unimodal language models (e.g. GPT2) whose outputs are conditioned directly on visual representations provide no discernible improvement in prediction, and that machine-generated linguistic descriptions of the stimuli explain a far smaller fraction (∼39%) of the explainable variance in ratings compared to vision alone. Taken together, these results showcase a general methodology for disambiguating perceptual and linguistic abstractions in aesthetic judgments using models that computationally separate one from the other.

1 Background

Aesthetic experience (the experience of beauty) is a universal phenomenon without a universal definition. Centuries of debate, from antiquity onwards, have asked why we experience beauty, and where it comes from [32, 36, 27, 23, 20, 12, 35, 29, 14, 38]. A central theme in these debates is the notion of ineffability: the extent to which our experience of beauty can be adequately described in natural language [16]. Given the inherent subjectivity of affective self-report, researchers have in many cases attempted to better operationalize ineffability by localizing or attributing our experience of beauty to various points along a somewhat Fodorian axis, which at one end assumes aesthetic experience is the product of a highly encapsulated process that is inaccessible to language and at the other assumes beauty is the product of conscious, deliberative, verbalizable thought [39, 33, 34, 28, 4].

These debates are challenging and difficult to arbitrate with behavior (i.e. empirical aesthetics) or neuroimaging (i.e. neuroaesthetics). In this work, we suggest that one potential route for moving this debate forward is with the use of computational models (i.e. computational aesthetics) in the form of deep neural networks [8, 3]. Deep neural network models trained on canonical computer vision...
and natural language processing tasks allow us to systematically control the kinds of computations and information processing mechanisms a given system can use to make inferences about aesthetic stimuli. Here, we use a linear decoding method to assess how well we can predict human ratings of beauty for a diverse set of naturalistic images from the features of unimodal and multimodal deep neural network models never trained explicitly on predictions of beauty. Our main goal in this is to better understand the relationship between representation learning and aesthetic experience, and how various task modalities modulate that relationship.

2 Methods

Our main source of human ratings in these experiments is the OASIS dataset [18], a set of 900 images curated to span a 7-point scale of arousal and valence ratings, and to which ratings of aesthetics were later added [5]. Each image comes with a rating that is the average of 100 to 110 human raters. To predict these group-average affect ratings, we use cross-validated regularized (linear) regression over features extracted from (pretrained) deep neural network models, none of which receive any prior training on aesthetic targets. To compute these regressions, we proceed layer by layer through each network, extracting the features and decoding the aesthetic ratings from these features in a procedure that mimics how information is decoded from neurophysiological signals. That is to say, we use each feature map to predict how subjects will rate an image, then correlate those predicted ratings with the actual ratings provided by the participants. The higher the correlation, the more information about aesthetics is available in a given feature map, with no more than a linear regression necessary to convert network activity into an aesthetic prediction. See Figure 1A and Appendix A.1 for details. The logic here is a logic of representational sufficiency: If the predictions of our feature regressions are accurate, it suggests that whatever the underlying computations producing aesthetics in the human brain may be, they need not be any more sophisticated than a single affine transformation of the kinds of representation produced by the feedforward, hierarchical operations of a deep neural network. In this analysis, we use this logic to probe what kinds of deep net representations are sufficient for predicting aesthetics, and better triangulate the computational pressures (i.e. tasks) that produce them. Note that the regression scores we report below take the form of what we call ‘explainable variance explained’. This is simply the squared Pearson correlation coefficient between predicted and actual ratings divided by the squared Spearman-Brown split-half reliability of the ratings across subjects (the ‘noise ceiling’). Note that given the quantity of subjects underlying the average, the noise ceiling for this data is extremely high at $r_{\text{Pearson}} = 0.988 \ [0.984, 0.991]$. Unless otherwise noted, we report the score of a model’s maximally predictive layer as that model’s overall score.

3 Results

Unimodal Vision Models Recapitulating previous work, we first show that pure unimodal vision models, in the form of contrastive (self-supervised) image models, are capable of predicting up to 75% of the explainable variance in the group-average beauty ratings. From a sample of 18 contrastive learning models that learn only over augmented image instances (e.g. Dino, SimCLR, SWaV), the average explained variance is 0.607 [0.566, 0.641]. The most predictive model, a RegNet64 trained using the SEER pretraining technique [11] explains 74.6% of explainable variance. While trained using roughly a billion images, this model’s representations are learned without symbolic or conceptual training targets (even in the form of the one-hot encoding vectors used to train object recognition models). This means that models trained on images alone can account for the majority of explainable variance in human beauty ratings.

Multimodal Vision Models The CLIP models [26] are a series of models trained on the task of linguistic alignment: given an image and a caption paired with that image, the model encodes both in an equidimensional latent space, computes the cosine similarity between them, then (during training) back-propagates any similarity less than 1 as a loss term. The representations of the visual encoder are thus directly shaped by language. OpenAI’s CLIP models (S/16, B/32, L/14, et cetera) all show substantive gains over the best-performing unimodal image model (RegNet64-SEER), with 80.5% to 87% of explainable variance explained.

\[^1\]This dataset contains no identifying information, and was cleared for public use by Harvard University’s Institutional Review Board.
Figure 1: A Schematic of our feature regression pipeline for decoding affective information from deep
net responses. Our target in these experiments are group-average beauty ratings, which we predict
by extracting image features from a candidate deep neural network model, (optionally) reducing
their dimensionality, then employing them as predictors in a cross-validated ridge regression with
the group-average beauty ratings as output. This method gives us a beauty decoding score per layer
per candidate model. B Results from our feature regression pipeline as applied to a unimodal vision
model (SimCLR), a purely language-aligned model (CLIP) and a model that combines unimodal
vision training and language alignment (SLIP) – all holding dataset and architecture constant. To
the left, we see results across layers (the semitransparent jagged lines are individual layer scores;
the curves are the output of a generalized additive smoother across layers). The takeaway here is
that for all models, predictive accuracy increases with depth. To the right, we see the results from
the maximally predictive layers of each model. Error bars are 95% confidence intervals across 1000
bootstrap resamples of the human subject pool. The takeaway here is that adding language alignment
(without taking away unimodal vision training) in the form of the SLIP objective does significantly
increase downstream readout of aesthetic information.

The problem with comparing the CLIP model directly to other models is that CLIP is trained on a
proprietary dataset of 400 million image-text pairs not yet available to the public. To address this
discrepancy, we use the SLIP models [22] – a series of Vision Transformers (Small [ViT-S], Base
[ViT-B], & Large [ViT-L]), all trained on the YFCC15M dataset (15 million image-text pairs), but on
only 1 of 3 tasks: pure SimCLR-style self-supervision; pure CLIP-style language alignment; or the
eponymous SLIP – a combination of self-supervision and language alignment. The SLIP models
allow us to control for the influence of language, holding architecture and dataset constant.

The pattern of results across the SLIP models suggests adding language to purely visual learning
does indeed increase the downstream predictive accuracy of aesthetic ratings. Specifically, while pure
CLIP-training shows discrepant gains over pure SIMCLR-training across the 3 vision transformer
sizes (performing slightly better in ViT-S and ViT-B, and slightly worse in ViT-L), SLIP-training
outperforms its pure SimCLR counterpart across all 3 transformer sizes by a significant, at least
midsize margin. A bootstrapping analysis using 1000 resamples of the human subject pool (averaging
across model size) shows the difference between SimCLR and CLIP to be nonsignificant, with a
bootstrapped mean of 0.0098 [-0.027, 0.041] (p = 0.67), while the difference between SimCLR and SLIP is significant, with a bootstrapped mean of 0.067 [0.037, 0.096] (p < 0.001). Results from these experiments are summarized in Figure 1B-C.

**Language Models via Captions** Adding language to visual representations by way of CLIP-style alignment does seem to facilitate better downstream prediction of aesthetic ratings. But what exactly is language doing here? Is it really just adding to the visual representation or is it changing that representation in some fundamental way? To assess this, we opted to test the outputs of a unimodal language model conditioned on CLIP’s visual encoder using our feature regression pipeline. This required first converting the visual embedding generated by CLIP into an embedding suitable for a language model. For this, we used a system called CLIP-Cap [21]. CLIP-Cap is a closed-loop system that employs a small multilayer perceptron (MLP) or transformer model to project the visual embedding from a CLIP model to a token embedding – called a ‘prefix embedding’ – that can be used by GPT2 [25] to generate a natural language caption.

![Figure 2: Results from our experiment using the CLIP-Cap model to assess how well the predictive power of CLIP’s visual outputs might be further compressed and translated into natural language. The red line in the facet on the right are the scores across the layers of the CLIP visual encoder used to generate an image ‘prefix’ embedding that is subsequently passed to GPT2 for captioning. The line in blue in the facet on the left is the predictive power of that prefix embedding as it is processed across the layers of GPT2. In other words, this line tracks the potential of GPT2 to facilitate better aesthetic decoding with the attention operations of natural language processing. The line in green is the predictive power of the generated caption passed back through GPT2 without the prefix embedding. This line tracks how well (machine-generated, image-conditioned) language alone might predict aesthetic ratings. The line in purple is the predictive power of the generated caption passed back through GPT2 with the prefix embedding. This line tracks whether visual embeddings and image-conditioned language together might outperform either one alone. The difference between the blue line and the green line represents the difference in predictive power between CLIP’s visual features and GPT2’s linguistic features – the difference, in other words, between language-aligned perception and language alone. This gap is substantial. The negative slope on the purple line seems to be an artifact of the feature regression overfitting to the embedding complexity added by the caption.](image)

For this experiment, we use CLIP-Cap’s MLP method of projection, which defaults to a prefix embedding length of 10 and uses CLIP-ViT-B/32 as its visual backbone. In the same way we decode aesthetics from features evoked by images in visual models, here we decode aesthetics from features evoked by the ‘prefix embeddings’ in the language model: that is to say, layer by layer, and using the same regression method. We find first and foremost that while the projected prefix embedding
Aesthetic experience is no single phenomenon, but a pluralistic combination of multiple different factors: our sensory and social ecologies, our bodies, our idiosyncratic developmental trajectories, our beliefs, and our perceptions [2, 54, 5]. An overarching goal of this and similar works is in some sense to approximate what percentage of aesthetic experience may be attributable to certain kinds of computational processes [4, 29, 8]. Here, we show that while perceptual processes in the form of feedforward, hierarchical, subsymbolic visual feature extraction are so far the best predictors of how people on average will rate the aesthetics of naturalistic image stimuli, language (alignment) may play a key role in shaping these representations. Furthermore, it seems that whatever the nature of the visual semantics that undergird the successful prediction of aesthetic responses in multimodal models like CLIP, at least a nontrivial portion of these semantics may be translated to machine-generated natural language descriptions. Aesthetic ineffability in this sense may be less of a binary and more of a gradient. The difference between the predictive power of an image in visual feature space and its description in natural language space could serve as a direct quantification of this gap.

Of course, this exact same point makes clear a few inherent limitations to some of the methods we’ve used here: simply put, not all image descriptions are made equal. Just as an expert orator may be more capable of evoking emotion with language than a novice, so too might certain descriptions communicate aesthetic value more effectively than others – even without explicitly affective qualifiers. Exposition of key details or interactions in a scene might be essential to communicating its aesthetic quality. To the extent that this is true adds immense complexity to the endeavor of disentangling vision from language, but the use of machine vision and language models does potentially allow us to pursue this disentanglement in ways that weren’t necessarily available to experimentalists before.

One immediate priority for future work is to assess the extent to which methods like consensus-based caption-scoring [37] could be used to reconcile divergent natural language descriptions of the same stimulus into a single representation – something that might allow us to supplement our machine-generated captions with crowdsourced human captions. Aggregating multiple natural language descriptions into a single coherent embedding might also be the key to closing the distance between visual representations and natural language descriptions that match these representations in terms of their downstream predictive power. Other, less proximate work should reconsider what it would mean for an affective experience (like the experience of beauty) to be communicated effectively between one agent and another, and whether this kind of communication has implications for learning.
The OASIS dataset is publicly available under a Creative Commons License at the following URL: https://osf.io/6pnd7/. All code will be made available upon publication. All experiments were run on a single Linux machine with 8 RTX3090 GPUs and 756GB of RAM. Most computations were CPU intensive and GPU use could be avoided entirely.

**Checklist**

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default [TODO] to [Yes], [No], or [N/A]. You are strongly encouraged to include a justification to your answer, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section 4.

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? YES (Data is publicly available; code will be released on publication).
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releases new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you’re using/curating? [Yes]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes]

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]
References


A Appendix

A.1 Method Details: Feature Regression

Our feature regression pipeline consists of 4 distinct phases: feature extraction; dimensionality reduction; ridge regression; cross-validation and scoring.

**Feature Extraction** We consider feature extraction from ‘every layer’ to mean the sampling of network activity generated after each distinct computational suboperation in a deep neural network model. This means, for example, that we consider a convolution and the nonlinearity that follows it as two distinct operations that produce two distinct feature spaces, both of which we consider candidates for decoding. If a layer returns a tensor with multiple components (such as a convolutional layer) we first flatten the tensor to a single component, such that the layer represents any given image as a feature vector. The layer thus represents a dataset of \( n \) images as an array \( F \in \mathbb{R}^{n \times D} \), where \( D \) is the dimensions of the feature vector.

**Sparse Random Projection** For some deep-net layers \( D \) is very large, and as such performing ridge regression directly on \( F \) is prohibitively expensive, with at best linear complexity with \( D \), \( O(n^2D) \) [13]. Fortunately it follows from the Johnson-Lindenstrauss lemma [15, 9] that \( F \) can be projected down to a low-dimensional embedding \( P \in \mathbb{R}^{n \times p} \) that preserves pair-wise distances of points in \( F \) with errors bounded by a factor \( \epsilon \). If \( u \) and \( v \) are any two feature vectors from \( F \), and \( u_p \) and \( v_p \) are the low-dimensional projected vectors, then;

\[
(1 - \epsilon)||u - v||^2 < ||u_p - v_p||^2 < (1 + \epsilon)||u - v||^2
\]

(1)

holds provided that \( p \geq \frac{4 \ln(n)}{\epsilon^2} \) [1]. With \( n = 900 \) for our dataset, to preserve distances with a distortion factor of \( \epsilon = .1 \) requires \( \geq 5830 \) dimensions. Thus we chose to project \( F \) to \( P \in \mathbb{R}^{n \times 5830} \) in instances where \( D >> 5830 \). To find the mapping from \( F \) to \( P \) we used sparse random projections following Li et al. [19]. The authors show a \( P \) satisfying (1) can be found by \( P = FR \), where \( R \) is a sparse, \( n \times p \) matrix, with i.i.d elements

\[
r_{ji} = \begin{cases} 
\sqrt{D} & \text{with prob. } \frac{1}{2\sqrt{D}} \\
0 & \text{with prob. } 1 - \frac{1}{\sqrt{D}} \\
-\sqrt{D} & \text{with prob. } \frac{1}{2\sqrt{D}} 
\end{cases}
\]

(2)

**Ridge Regression with LOOCV** We used regularized (ridge) regression to predict the average human ratings of images, \( Y \), from their associated (dimensionality-reduced) deep net features, \( P \). As our goal was not to identify a particular regression model for later use, but rather get a best estimate for the linear read-out of beauty scores from deepnet feature spaces, we utilized all the data at our disposal with a leave-one-out (generalized) cross-validation procedure. For every image in our dataset
(∀i ∈ {1...900}) we fit the coefficients $\hat{\beta}_i$ of a regression model on the remaining data, such that

$$Y_{-i} = P_{-i}\beta_i + \epsilon$$

with minimal $\|\epsilon\|$ (error). Ridge regression penalizes large $\|\beta\|$ proportional to a hyper-parameter $\lambda$, which is useful to prevent overfitting when regressors are high-dimensional (as with $P$). We first standardized $Y$ and the columns of $P$ to have a mean of 0 and standard deviation of 1. Let $P_{-i}$ and $Y_{-i}$ denote $P$ and $Y$ with row $i$ missing, then each $\hat{\beta}_i$ is calculated by;

$$\hat{\beta}_i = \left((P'_{-i}P_{-i} + \lambda I_p)^{-1}P'_{-i}Y_{-i}\right)$$ (3)

Each $\hat{\beta}_i$ is then used to predict the beauty rating from the deepnet feature projection of each left out image;

$$\hat{y}_i = P_i\hat{\beta}_i, \quad \tilde{Y} = \{\hat{y}_i\}_{i=1}^{900}$$ (4)

The hyper-parameter $\lambda$ we set at $1e4$, a value we determined using a logarithmic grid search over $1e-1 - 1e6$ on an AlexNet model that we subsequently exclude from the main analysis. $\lambda = 1e4$ yielded the smallest cross-validated error ($\|Y - \tilde{Y}\|$) when averaging across layers. We used the RidgeCV function from [24] to implement this cross-validated ridge regression, as its matrix algebraic implementation identifies each $\hat{\beta}_i$ in parallel, resulting in significant speedups [31].

Scoring In this analysis, we score each deepnet layer by computing the Pearson correlation coefficient between its predicted ratings, $\tilde{Y}$, and the actual group-average affect ratings from the human subjects, $Y$. To convert this Pearson correlation coefficient into a score that represents the percentage of explainable variance explained, we divide the square of this coefficient by the square of the Spearman-Brown splithalf reliability that constitutes the noise ceiling.

Note that previous empirical work suggests the sparse random projection step in this pipeline is largely optional and can, without substantial decrease in accuracy, be eliminated in favor of directly using the full-size, flattened feature maps in the regression.

A.2 Caption Accuracy + Better Language Models

A major issue with the use of machine-generated captions in any pipeline is their accuracy. Even state-of-the-art captioning models make consistent, common-sense errors no human would make in describing an image [40]. What does this mean for the current experiment?

One point to consider is that we are not necessarily interested in the accuracy of the caption per se, but the extent to which that caption reflects the information content available in the visual embeddings of CLIP, which themselves may not accurately reflect category-level or more generally semantic content.

The issue then is not whether CLIP-Cap (or other systems that interpret CLIP’s visual embeddings in service of caption generation, such as Cho et al. [7]) provides accurate human-legible captions, but whether those captions reflect a coherent summary function of CLIP’s visual embeddings. This is admittedly difficult to measure, but because CLIP-Cap and similar models are gradient-based, we can say definitively, at least, that the resultant captions are literal functions of CLIP’s vision.

This somewhat oblique approach here obviously invites the question of why, instead of machine-generated captions, we do not simply use human-generated captions. The answer is twofold: First, because we are interested in more directly interpreting CLIP’s latent space, this makes human captions mostly irrelevant. Second, and perhaps more importantly, though, is the issue of how we sift through, concatenate or rank across the varied captions that human subjects will invariably provide. Given 5 sentence-level descriptions of an image, what is the numerical vector that summarizes them and allows us then to regress this vector onto the aesthetic rankings provided for those images? It is possible that consensus-based methods [37] provide a reasonable answer here, but this remains an open question that will require further work beyond the scope of this analysis.

Another potential issue with the use of machine-generated captions specifically in this pipeline are the large language models we use to transform those captions into embeddings appropriate for our feature regression pipeline. CLIP-Cap uses as its language transformer a standard (midsize) GPT2 model. Language models are known to be far more accurate with scale [17]. Could other language models (in conjunction with better captions) facilitate greater decoding accuracy?
While by no means an exhaustive experiment, we explored this question preliminarily by expanding our caption-based decoding paradigm to two other sets of captions and two other large language models. For captions, we considered CLIP-Caption-Reward [7] (another CLIP-based caption-generation algorithm that uses CLIP similarity as a reward function) and GIT (Generative Image-to-Text Transformer) [40]. For language models, we considered GPT2-XL (the larger version of the GPT2 used by CLIPCap for caption generation) and the All-MPNet-Base-V2 variant of S-BERT [30] (the largest thereof). While no single caption and model combination exceeds 58% of explainable explained variance (compared to the visual encoder’s 82%), the best combination (SBERT-over-GIT captions), improves nearly 20% over the baseline we test in the main results (GPT2-over-CLIPCap) at 38.5%. This latter caption-model combination notably does not involve CLIP, which makes it irrelevant as a method of interpreting the CLIP visual encoder’s predictive accuracy, but it does suggest one potential route forward for assessing the impact of language on aesthetic judgment.

Figure 3: Results from our experiment extending our feature regression pipeline for aesthetics decoding from captions beyond the CLIP-Cap model. The bar at the top of the plot is the best performing layer from a CLIP visual encoder operating over images. The bar beneath it (GPT2-over-CLIPCap) is the result from the main experiment detailed in Section 3. Error bars are 95% confidence intervals over 1000 bootstrap resamples of the human subject pool.