Shared Visual Representations of Drawing for Communication: How do different biases affect human interpretability and intent?

Anonymous Author(s)
Affiliation
Address
email

Abstract

We present an investigation into how representational losses can affect the drawings produced by artificial agents playing a communication game. Building upon recent advances, we show that a combination of powerful pretrained encoder networks, with appropriate inductive biases, can lead to agents that draw recognisable sketches, whilst still communicating well. Further, we start to develop an approach to help automatically analyse the semantic content being conveyed by a sketch and demonstrate that current approaches to inducing perceptual biases lead to a notion of objectness being a key feature despite the agent training being self-supervised.

1 Introduction

Recently Mihai & Hare (2021b) demonstrated that it was possible to train a pair of agents parameterised by artificial neural networks to play a visual communication game by communicating through line drawings. Such a demonstration was often discussed as a logical next step in the emergent communication community but was challenging until the innovation of a differentiable rasterisation algorithm (Mihai & Hare, 2021a) that allowed end to end training. The referential signalling games, inspired by Lewis (1969), involved a 'receiver' agent that was presented with a set of images, and a 'sender' agent that was presented with a single image from the receiver's set. The goal of the game was for the sender agent to communicate to the receiver which image they had by drawing a picture using a fixed number of line strokes. Without specific inductive biases, it was demonstrated that successful communication could be achieved between the agents, however, the images themselves were not interpretable; they were in essence the visual equivalent of a hash code. Mihai & Hare (2021b) however, went on to further demonstrate that, with appropriate representational biases, induced by additional loss functions during training, there was strong evidence that the drawings produced by the machine could also be used to communicate successfully with humans.

In this paper we explore Mihai & Hare (2021b)'s model in more detail. We start by introducing a more powerful network for encoding visual information — the pretrained Vision Transformer (Dosovitskiy et al., 2021) from the CLIP framework (Radford et al., 2021), and then explore different approaches to inducing the network to produce more understandable drawings. We compare against the pretrained VGG16 feature extractor used in the original model and develop an approach that enables us to ask what the main semantic content of the drawings are using "prompt engineering" (Radford et al., 2021) with the CLIP model.

Submitted to 3rd Workshop on Shared Visual Representations in Human and Machine Intelligence (SVRHM 2021) of the Neural Information Processing Systems (NeurIPS) conference.

2 Background

Recent works on visual communication games (Mihai & Hare, 2021b; Fernando et al., 2020), which are analogous to the approach taken by emergent communication research (Havrylov & Titov, 2017; Lazaridou et al., 2017, 2018; Bouchacourt & Baroni, 2018) have shown that agents can successfully convey information by drawing, which is a much simpler and more interpretable form than language. The main challenges with a language-based communication protocol developed between agents playing such games are its interpretability and grounding into natural language, which would allow a human to understand and use the information conveyed (Lowe et al., 2019). Drawing and sketching, on the other hand, has been used since prehistoric times by humans to depict the surrounding visual world, long before the emergence of written language (Gelb, 1963). Sketches of a visual scene produced by a human could have various levels of abstraction: from very detailed and close to reality representations, to depictions of just the semantics expressed in terms of the objects in the scene and their relations, to low-level representations like edges and key points. Works from Neuroscience suggest that simple line drawings capture the core visual features needed to recognise physical objects (Biederman & Ju, 1988) and activate the same brain regions used to distinguish natural scene categories from colour photographs (Walther et al., 2011). Marr's classical theory of vision (Marr, 1982) also posited that sketch-like representations were a fundamental part of human vision.

3 Extended Model with CLIP

In this work, we follow the game setup proposed by Mihai & Hare (2021b) which was inspired by Havrylov & Titov (2017)'s image guessing game. As illustrated in Figure 1, the game requires the sender to communicate the target image to the receiver, by sketching 20 black and white straight lines. The receiver has to guess the correct image from a pool of photographs consisting of K distractors plus the target. An overview of the agents' architecture and game setup is shown in Figure 1. In the "original" game setup, the photograph that the sender communicates about matches the target from the receiver's pool of images. Mihai & Hare (2021b), however, proposed other game setups in which this requirement does not hold. For the purpose of this study, we only explore the original game variant for which we set the number of distractors, K, to 99. It is worth noting that this sort of game would be very difficult for humans to play. Guessing the target from a set of 100 images, which could contain multiple examples from the same class as the target, based only on a 20-line black and white sketch seems impossible for humans. The trained agents, however, manage to establish a visual communication protocol that can be used to successfully solve the task as shown in Section 3.2.

3.1 Inducing interpretable drawings with "perceptual" losses

Zhang et al. (2018) demonstrated that a loss computed using the weighted difference of features extracted across a range of low and intermediate layers in a pretrained VGG-16 (Simonyan & Zisserman, 2015) and AlexNet (Krizhevsky et al., 2012) CNN could predict human perception of the similarity of images. Building upon this idea Mihai & Hare (2021b) demonstrated that such a loss function could be used to induce a drawing agent to produce sketches that were significantly more interpretable by humans (in the sense of improved agent-human gameplay) than agents trained without such a loss. In the latter case, the agents could learn to play the game well and generalise to unseen images, but the drawings produced by the sender were essentially visual representations of hash codes with rather random sets of lines.

With our modification to move from the VGG16 feature extraction network to the Vision Transformer (ViT) model of the CLIP framework (Radford et al., 2021), we first explored whether a similar type of perceptual loss to that used by Mihai & Hare (2021b) would be possible. We extracted the feature after each transformer residual block from both the sketch produced by the sender and the corresponding photo that was presented to the sender and used this to compute the loss. The loss itself involves normalising each layer's features, computing the sum squared difference between sketch ($\boldsymbol{S}$) and image ($\boldsymbol{I}$) features at each layer l and performing a weighted sum over the layers, L,

$$\mathrm{l}_{\mathrm{perceptual}}(\boldsymbol{S}, \boldsymbol{I}, \boldsymbol{w}) = \sum_{l \in L} \frac{\boldsymbol{w}_l}{n_l} \left\| \hat{\boldsymbol{S}}^{(l)} - \hat{\boldsymbol{I}}^{(l)} \right\|_2^2 , \tag{1}$$

where n_l is the dimensionality of the l-th layer feature. We used fixed uniform weights for each layer in experiments presented here, $w_l = 1 \, \forall l \in L$.

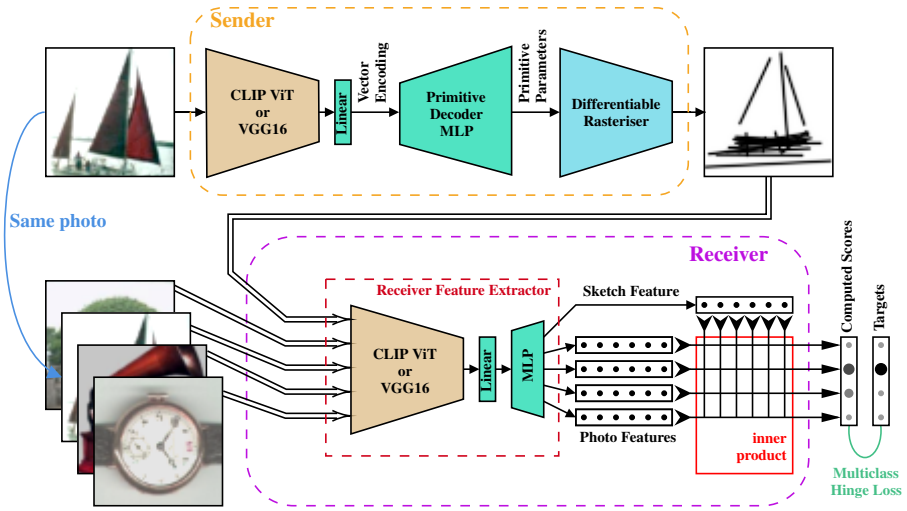

Figure 1: **Model overview.** Two agents are trained to play an image guessing game in which they communicate through a simple line drawing. As indicated, unlike in Mihai & Hare (2021b)'s original model, we also experiment with a more powerful pretrained Vision Transformer encoder module (ViT-B/32 from CLIP (Radford et al., 2021)). An additional perceptual loss (not shown) between the sender's input photo and output sketch induces the sketch to be more understandable. Figure adapted from Mihai & Hare (2021b).

.

We also investigate another method for generating interpretable drawings, inspired by the approach taken by Frans et al. (2021) The key idea is that we add an extra loss, $\mathrm{l}_{\mathrm{clipdraw}}$, computed by the cosine distance (negative cosine similarity) between the encoded representation, $f(\cdot)$ (e.g. from the last layer of the ViT encoder, or `relu5_3` of the VGG16) of the generated sketch and input image. However, such a loss alone does not result in sketches that are perceptually similar to the input, so instead perceptual similarity is induced by computing the loss over a set of randomly *transformed* sketches, T:

$$\mathrm{l}_{\mathrm{clipdraw}}(\boldsymbol{S}, \boldsymbol{I}) = -\sum_{t \in T} \frac{f(t(\boldsymbol{S})) \cdot f(\boldsymbol{I})}{\|f(t(\boldsymbol{S}))\| \|f(\boldsymbol{I})\|} \ . \tag{2}$$

Following Frans et al. (2021), T consists of four randomly sampled transformations created by applying a random perspective transformation and random resizing and cropping in sequence. This crude modelling of physical spatial constraints is sufficient to induce the sketches to be interpretable.

3.2 Results

We next present results of the visual communication game played with STL-10 images (Coates et al., 2011) in the original game setup as described by Mihai & Hare (2021b). Figure 2 shows test communication success rates and sketches produced by models constructed with either a VGG16 or ViT image encoder, trained with only the game objective $\mathrm{l}_{\mathrm{game}}$, or with the addition of either of the two perceptual losses, $\mathrm{l}_{\mathrm{perceptual}}$ and $\mathrm{l}_{\mathrm{clipdraw}}$, described in Section 3.1. For the experiments run with the ImageNet-pretrained VGG16 image encoder, we use the same parameters as specified in Mihai & Hare (2021b). When replacing the image encoder with CLIP's pretrained ViT-B/32 model (Radford et al., 2021), we found that increasing the hidden sizes of the Primitive Decoder, shown in Figure 1, from 64 and 256 to 1024 each, significantly improves the quality of the sketches. It is also worth noting that a bigger learning rate is needed for this model to converge, more specifically 0.001.

Results show that models trained with only the game objective achieve the highest communication success rate, i.e. agents can successfully communicate about the target image, although, they do so by drawing what it looks to us like random sets of lines. The addition of perceptual losses to $\mathrm{l}_{\mathrm{game}}$ leads to significantly more interpretable sketches. The model using CLIP'S image encoder, pretrained on the task of matching (image, text) pairs, leads to better image representations, and eventually, sketches which can be much easily interpreted by humans than those produced by a model with a

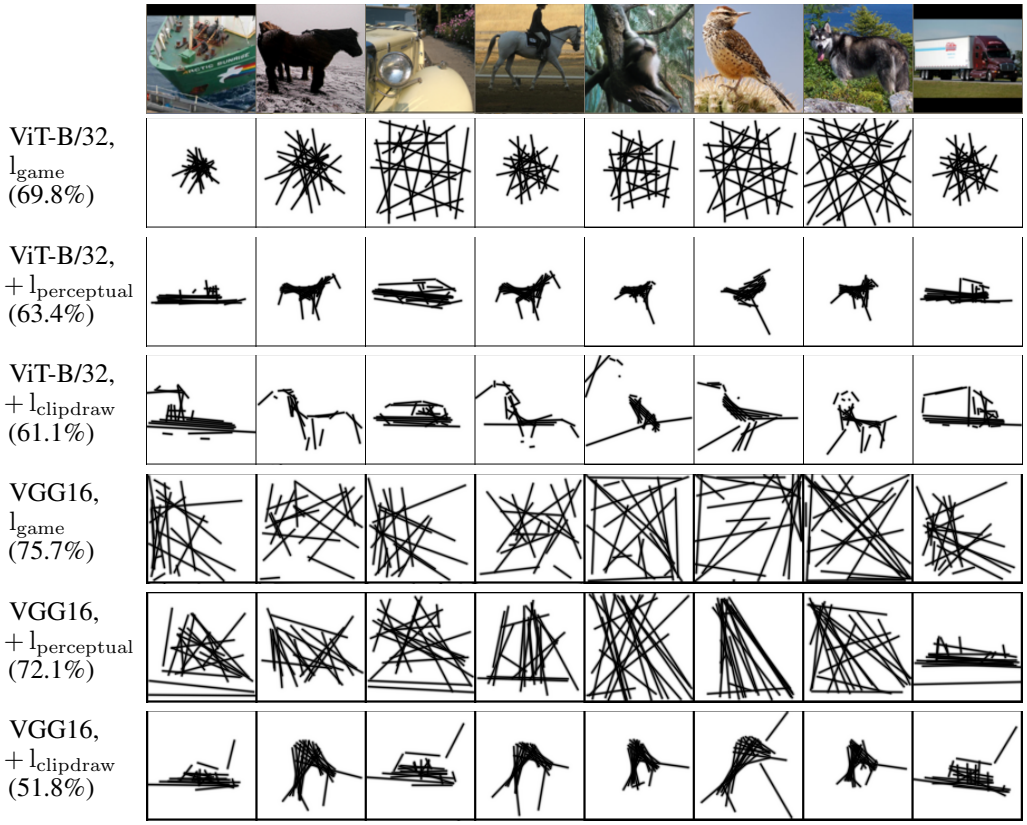

Figure 2: **Sketches from the visual communication game using STL-10 dataset with different image encoders and "perceptual" losses**. Models trained with the l_{game} only do not learn to draw in an interpretable fashion. For both ViT-B/32 and VGG16 image encoders, the addition of either perceptual loss induces more structure into the resulting drawings, making them more similar to the subject of the image, although it decreases that agents' communication success (shown in brackets on the left side).

VGG16 encoder pretrained for the supervised image classification task on ImageNet. More sketches generated during testing can be found in Appendix B.

4 Exploring what the drawings mean with prompt engineering

It would be beneficial to be able to understand what information the sender agent is trying to convey through its sketch and compare that to what a human playing the game might try to impart. In the particular game setting we are using, if one communicates only the object in the scene then the expected communication rate would only be 10%. To achieve higher rates much more nuanced information about the image contents needs to be conveyed. Ultimately understanding what is being communicated, and how it differs to humans would allow us to design better approaches to inducing more human-like behaviour in the model and the agent's internal representations.

To start to explore this in more detail, we demonstrate that we can begin to answer the question of what is being communicated by using the CLIP model as a probe. With the technique of prompt engineering, where a set of textual prompts are encoded with CLIP's language model, it becomes possible to ask basic questions about how CLIP perceives a sketch in terms of the semantic content. For the initial experiments presented here we use two prompt templates: ''`a drawing of a XXX.`'' and ''`a photo of a XXX.`''. The placeholder (`XXX`) is replaced by the 10 different classes in the STL-10 dataset to create a complete set of 20 prompts. For each of the models we then compute a number of statistics regarding CLIPs perception of the sketch, the target image (i.e. the receiver image that is the true answer) and the guessed image (the image that the receiver actually picked),

averaged over all 8000 possible games in the STL-10 test set. More specifically we ask which class CLIP perceives an image $\boldsymbol{I}$ to be, using a function $\mathrm{c}(\boldsymbol{I})$ which returns the placeholder of the closest prompt (using cosine similarity in the embedding space), and compare CLIPs predicted class between the sketch, guess and target. In a similar way, we also compute which of the two templates `{photo, drawing}` CLIP predicts the sketch, target and guess to belong to. Finally, we also utilise a function $\mathrm{gt}(input)$ which returns the STL-10 ground-truth class label of the sender agent's input, to allow us to analyse to what extent CLIP's perception of the images matches the true label. The results of this analysis are shown in Table 1.

Table 1: **Comparing models with CLIP using prompt engineering.** There are significant differences between models, however, it is clear that the perceptual losses strongly encourage a more object-centric representation. CLIP is very good at telling the difference between sketches and photos in all cases, despite the perceptual losses pulling together the representations.

	VGG16 encoder			ViT-B/32 encoder		
	$\mathrm{l}_{\mathrm{game}}$	$+\mathrm{l}_{\mathrm{perceptual}}$	$+\mathrm{l}_{\mathrm{clipdraw}}$	$\mathrm{l}_{\mathrm{game}}$	$+\mathrm{l}_{\mathrm{perceptual}}$	$+\mathrm{l}_{\mathrm{clipdraw}}$
c(sketch)==gt(input)	7.3%	24.0%	41.2%	9.4%	96.4%	96.6%
c(sketch)==c(target)	7.3%	24.2%	41.5%	9.7%	96.4%	96.5%
c(sketch)==c(guess)	7.6%	24.6%	38.8%	10.4%	94.9%	96.1%
c(target)==gt(input)	97.3%	97.3%	97.3%	97.3%	97.3%	97.3%
c(guess)==gt(input)	85.6%	82.1%	76.5%	77.6%	94.8%	95.8%
tp(sketch)=='drawing'	100%	99.9%	99.9%	99.9%	96.2%	99.3%
tp(target)=='photo'	99.4%	99.4%	99.4%	99.4%	99.4%	99.4%
tp(guess)=='photo'	99.4%	98.4%	98.4%	99.0%	99.4%	99.4%

The results in Table 1 indicate that both forms of perceptual loss do a good job of making the sender agent produce sketches that capture the main class of object in the input image. There is an inherent bias towards CLIP generated sketches because the same model is being used to perform the generation and the probing. However, as can be seen in Figure 2 the CLIP generated sketches are qualitatively more interpretable than the VGG ones. When looking at these results bear in mind that the communication game itself is entirely self-supervised; the notion of object class is clearly not required for successful communication and is instead a side effect of inducing a perceptual loss between internal representations. The results also show that despite the perceptual losses forcing the representations of the sketch and image together, the CLIP-based probe is able to recognise the sketch as being a drawing and the receiver images from the dataset as being photos almost all of the time.

5 Discussion and Future Directions

The aim of this study was to explore how representational losses influence the sketches produced by artificial agents playing a visual communication game. We showed that the addition of either perceptual loss to the communication game leads to qualitatively more recognisable sketches than those produced by agents trained with the game objective only. Although the additional representational losses slightly decrease the agents' ability to communicate, they significantly increase the possibility to recognise the sketches as the semantic category of the photographs they represent (see in Table 1). Going forwards it would be interesting to explore other loss formulations, and see if it is possible to minimise the drop in communication rates that arise from introducing this perceptual bias.

Our brief experiments with prompt engineering in the previous section open up a number of doors for future analysis. The game being played by the agents is complex, and the communication success rates are far in excess of the 10% that would naïvely result from the models only communicating information about the class. The obvious next step is to question what additional information is being conveyed in the sketches; is it interpretable semantic information about the input image, or is it some kind of neural hash code that just happens to allow communication to succeed, or is it a mixture of both aspects? Further, similar to the causal interventions that are applied in emergent communication scenarios with non-visual channels, we would also wish to explore which parts of a sketch (perhaps bundles of strokes) contribute to particular aspects of semantic meaning. We hope that with richer datasets and considerably more engineering of prompts that the setup we've outlined in this paper would allow these goals to be achieved.

References

Irving Biederman and Ginny Ju. Surface versus edge-based determinants of visual recognition. *Cognitive psychology*, 20(1):38–64, 1988.

Diane Bouchacourt and Marco Baroni. How agents see things: On visual representations in an emergent language game. *arXiv preprint arXiv:1808.10696*, 2018.

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 215–223. JMLR Workshop and Conference Proceedings, 2011.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021.

Chrisantha Fernando, Daria Zenkova, Stanislav Nikolov, and Simon Osindero. From language games to drawing games. *arXiv preprint arXiv:2010.02820*, 2020.

Kevin Frans, Lisa B. Soros, and Olaf Witkowski. Clipdraw: Exploring text-to-drawing synthesis through language-image encoders. *CoRR*, abs/2106.14843, 2021. URL `https://arxiv.org/abs/2106.14843`.

Ignace J Gelb. *A study of writing*. University of Chicago Press, 1963.

Serhii Havrylov and Ivan Titov. Emergence of language with multi-agent games: Learning to communicate with sequences of symbols. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 2149–2159. Curran Associates, Inc., 2017.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012.

Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. Multi-agent cooperation and the emergence of (natural) language. In *International Conference on Learning Representations*, 2017.

Angeliki Lazaridou, Karl Moritz Hermann, Karl Tuyls, and Stephen Clark. Emergence of linguistic communication from referential games with symbolic and pixel input. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=HJGv1Z-AW`.

David K. Lewis. *Convention: A Philosophical Study*. Wiley-Blackwell, 1969.

Ryan Lowe, Jakob Foerster, Y-Lan Boureau, Joelle Pineau, and Yann Dauphin. On the pitfalls of measuring emergent communication. *arXiv preprint arXiv:1903.05168*, 2019.

David Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Henry Holt and Co., Inc., New York, NY, USA, 1982. ISBN 0716715678.

Daniela Mihai and Jonathon S. Hare. Differentiable drawing and sketching. *CoRR*, abs/2103.16194, 2021a. URL `https://arxiv.org/abs/2103.16194`.

Daniela Mihai and Jonathon S. Hare. Learning to draw: Emergent communication through sketching. *CoRR*, abs/2106.02067, 2021b. URL `https://arxiv.org/abs/2106.02067`.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

Dirk B Walther, Barry Chai, Eamon Caddigan, Diane M Beck, and Li Fei-Fei. Simple line drawings suffice for functional mri decoding of natural scene categories. *Proceedings of the National Academy of Sciences*, 108 (23):9661–9666, 2011.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

Checklist

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? [N/A]
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)? [No] however, we will make code available if accepted.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix A.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] this was impractical given the amount of compute required, but tests confirmed that variance from initialisation is small, mirroring what was reported in the detailed study in Mihai & Hare (2021b) who's model we adapt.
   (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] See Appendix A.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] We build off the code of (Mihai & Hare, 2021a,b) and Radford et al. (2021). We perform experiments with the STL-10 dataset (Coates et al., 2011).
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No] See 3a.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
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]

A Model Setup and Training details

For experiments with the VGG16 feature extractor, our setup exactly mirrors Mihai & Hare (2021b):

- STL-10 images are presented to the model at 96x96 resolution
- The final layer VGG-16 features are flattened and projected into a 64-dimension latent space in both sender and receiver agents.
- The sender agent uses an MLP with hidden sizes of 64 and 256 and output layer size of 80 (corresponding to the coordinates end-points of the 20 lines).

- The receiver agent uses an MLP with a 64 dimensional hidden layer and 64 dimensional output.
- all hidden layers use ReLU activations.
- the weights of the VGG16 feature extractor are fixed.
- Training is done with Adam using and initial learning rate of 1e-4.

For our CLIP-ViT-B/32 modification, VGG16 is replaced with the ViT and everything else remains the same except:

- STL-10 images are rescaled and presented to the model at 224x224 resolution to match the size the ViT was trained on. Both qualitatively and quantitatively making this change for the VGG16 model did not have much effect beyond slowing down training and inference.
- The sender agent uses an MLP with hidden sizes of 1024 and 1024 and output layer size of 80 (corresponding to the coordinates end-points of the 20 lines). We found the model would not work well with the sizes used for the VGG16; conversely training the VGG16 with this increased size didn't qualitatively change the sketches very much, nor affect the communication rate.
- the weights of the ViT-B/32 feature extractor are fixed.
- Training is done with Adam using and initial learning rate of 1e-3. Using the higher rate seems to help speed up convergence with the ViT. Conversely, with the VGG16 the higher initial learning rate leads to a model that doesn't converge.

B Additional Sketching Examples

In Figures 3 and 4 we provide additional examples to better illustrate the difference between the different encoder networks and the effect of including additional perceptual losses on the resulting sketches.

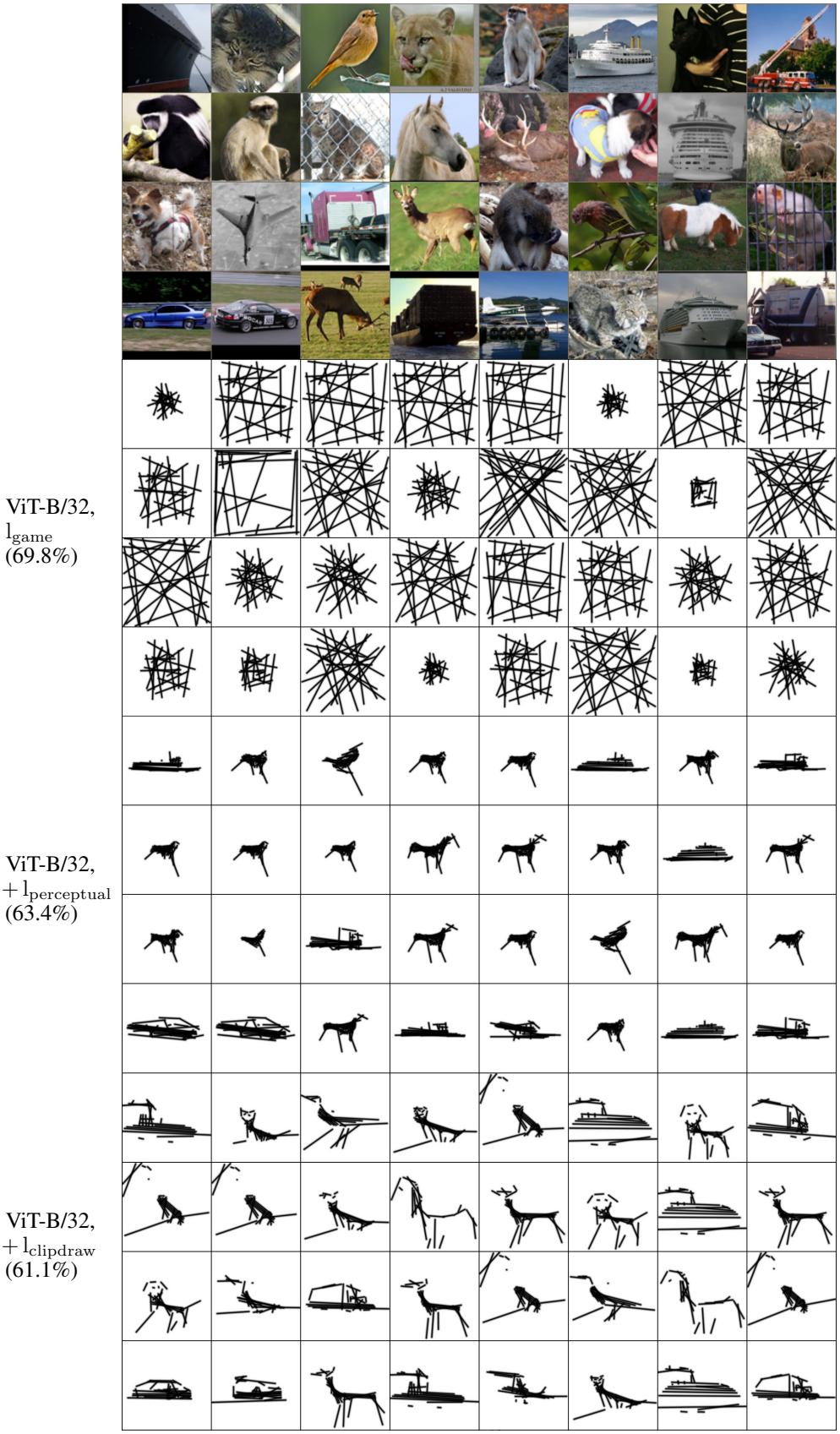

Figure 3: **More sketches of STL-10 test images produced by the model with ViT-B/32 encoder.**

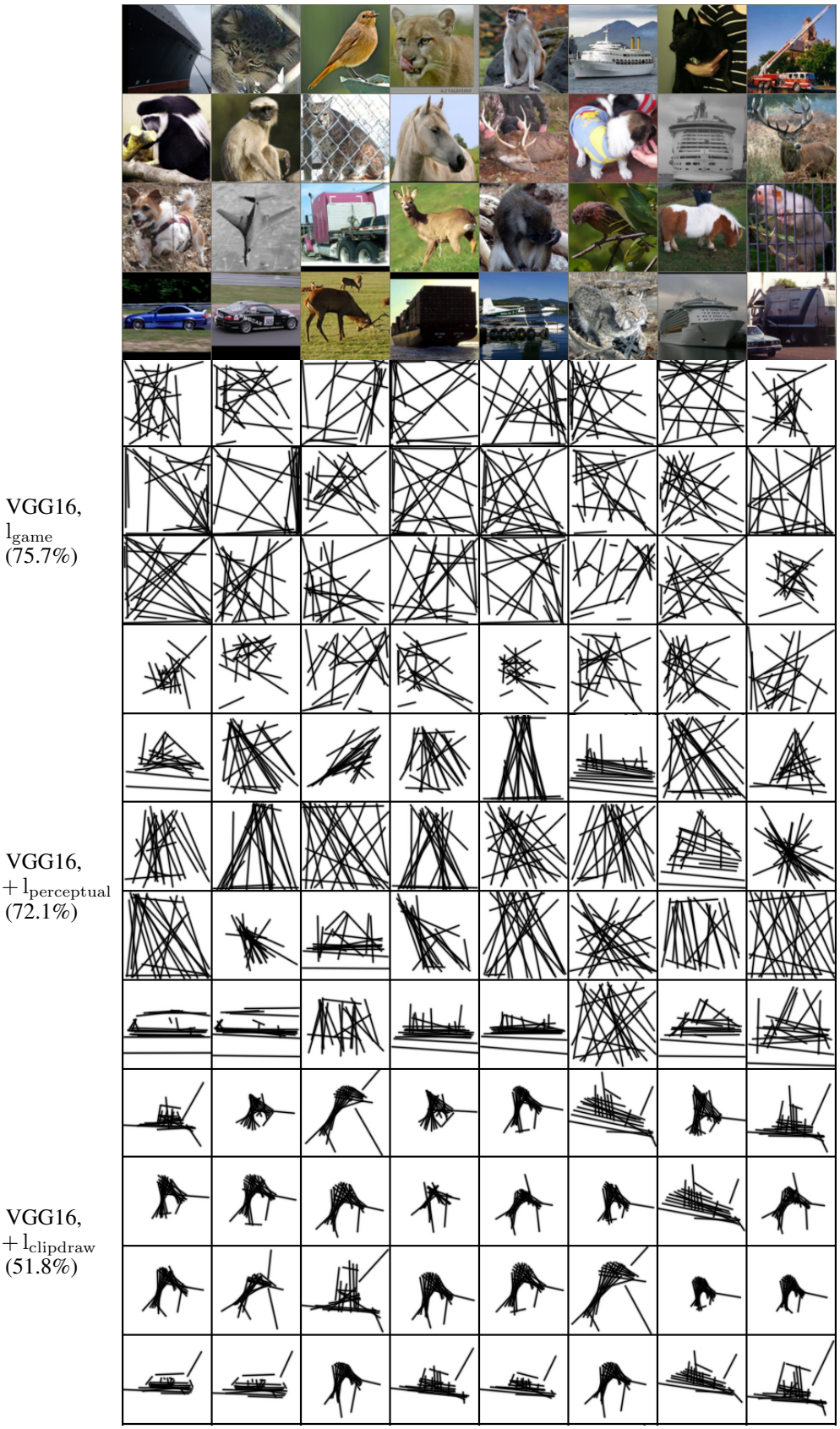

Figure 4: **More sketches of STL-10 test images produced by the model with VGG16 encoder.**