Finding Maximally Informative Patches in Images

Howard Zhong
MIT CSAIL
howardzh@mit.edu

Guha Balakrishnan
MIT CSAIL
balakg@mit.edu

Richard Strong Bowen
Cornell Tech
rsb@cs.cornell.edu

Ramin Zabih
Cornell Tech
rdz@cs.cornell.edu

William T. Freeman
MIT CSAIL
billf@mit.edu

Abstract

We consider the problem of distilling an image into an ordered set of maximally informative patches, given prior data from the same domain. We cast this problem as one of maximizing a pointwise mutual information (PMI) objective between a subset of an image's patches and the perceptual content of the entire image. We take an image synthesis-based approach, reasoning that the patches that are most informative would also be most useful for predicting other pixel values. We capture this idea with an image completion CNN trained to model the PMI between an image's perceptual content and any of its subregions. Because our PMI objective is a submodular, monotonic function, we can greedily construct patch sets using the CNN to obtain a provably close approximation to the intractable optimal solution. We evaluate our approach on datasets of faces, common objects, and line drawings. For all datasets, we find that a surprisingly few number of patches are needed to reconstruct most images, demonstrating a particular type of redundancy of information in images, and new potentials in their sparse representations. We also show that these minimal patch sets may be used effectively for downstream tasks such as image classification.

1 Introduction

Natural image datasets often lie on surprisingly low-dimensional manifolds due to significant correlations between image pixels and across images [40]. As a result, the pixel domain tends to be a highly redundant representation for many vision tasks, where only a subset of a scene needs to be observed to ensure success. This is illustrated by "attention"-based classifiers that focus on image subregions [45, 49], and cognitive science studies on "minimal" images for recognition [3, 44]. Put another way, by just observing a subset of an image's pixels, we can often *infer more about the image than just those pixels*. Studying what in an image is predictable given a very limited view of it reveals the regularities within and across images, and can influence how we choose to store or process them.

This paper reveals the strength of those image regularities through the following task: given an image I, a fixed patch size, and training images from the same domain, sequentially determine which patch in I provides the most new information about I. Consider the example in Fig. 1. Suppose we know that the image is in the CelebaHQ dataset, consisting of faces of various celebrities. Our approach will sequentially select patches coinciding with informative regions — areas both within and outside the face — that help to infer the full image. In addition to revealing fundamental properties of image collections, the answer to this question has potential impact in the design of efficient vision algorithms, and may relate to human cognition studies of attention and foveation.

NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications

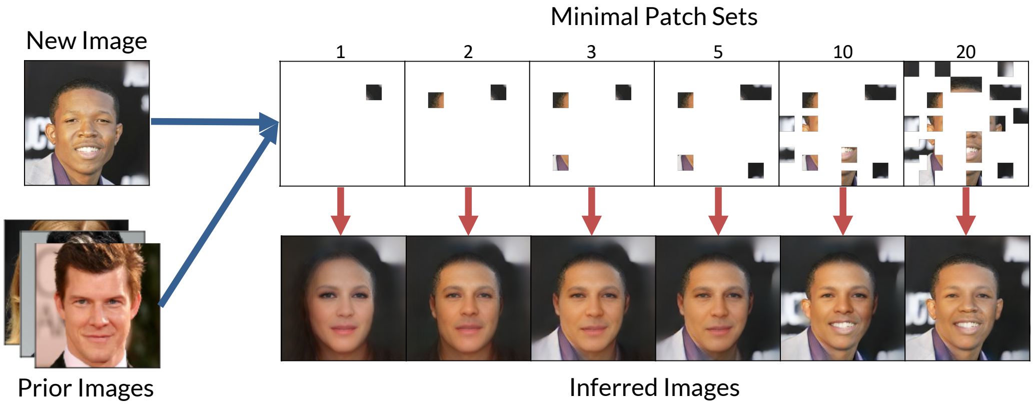

Figure 1: **Illustrative example of selecting informative image patches.** We aim to take an input image, and find the set of patches that are best able to help us reconstruct the image, assuming other data from the domain as prior knowledge (training images). The optimal patch sets and synthesized images in this figure are produced by our method.

We propose a method for this task that leverages a combination of probabilistic modeling, image synthesis, and submodular function optimization. We first propose a measure of pointwise mutual information (PMI) between an image and any subset of its patches with a probabilistic notion of perceptual similarity. We then design a convolutional neural network that yields an approximation of this PMI measure for a data domain of interest. Given a subset of observed image patches as input, this network synthesizes a completed image, along with uncertainty estimates in perceptual space. We then use this network to sequentially find maximally-informative patches for a test image in a greedy manner, which is an efficient and provably good approximation to the optimal solution.

We evaluate our work on several real datasets with different characteristics: face images (Celeba-HQ [23, 29]) , common objects (STL-10 [9]) and line drawings (QuickDraw [16]). We show that, given training data from a particular domain, surprisingly few image patches need to be observed to reconstruct the full image. We also explore how this finding may be useful for performing downstream tasks like classification.

2 Related Work

Our computational work has ties to several areas in cognition, the closest being visual attention. Humans selectively attend to parts of the visual space to acquire information, and combine information from these views to form an internal representation of the scene [33]. Attention has been studied in relation to human reasoning [4, 10], saccades and eye movements [12, 13, 18], motor control [1, 5], and peripheral vision [35]. Our work also has ties to findings on "minimal images," the smallest regions of an image that remain recognizable for humans [3, 44]. Finally, games like *Bubbles* and *Peekaboom* were developed to use human attention to annotate salient regions of image data in an interactive way [11, 47]. We draw inspiration from these works, though our method is purely data-driven and not strongly based on the cognitive system.

Inspired by the human cognitive system, computer vision researchers have incorporated attention into various learning models. Several studies use attention to localize features for fine-grained image classification [14, 48, 49, 51] and for image captioning and visual question answering tasks [2, 8, 31]. Attention can also provide computational savings for image processing, by isolating interesting regions to explore further [30]. In contrast to these studies, we focus on understanding how little of an image must be attended to to understand its contents.

Mutual information and submodular function optimization are topics with wide applications in AI and other fields. They have had a few uses in computer vision too. Mutual information has been used for registration between pairs of images [15, 36, 37, 46], and pointwise mutual information was proposed for grouping pixels within an image [19]. Submodular function optimization has been used to measure image region saliency and to segment image regions [20, 21, 25]. These works do

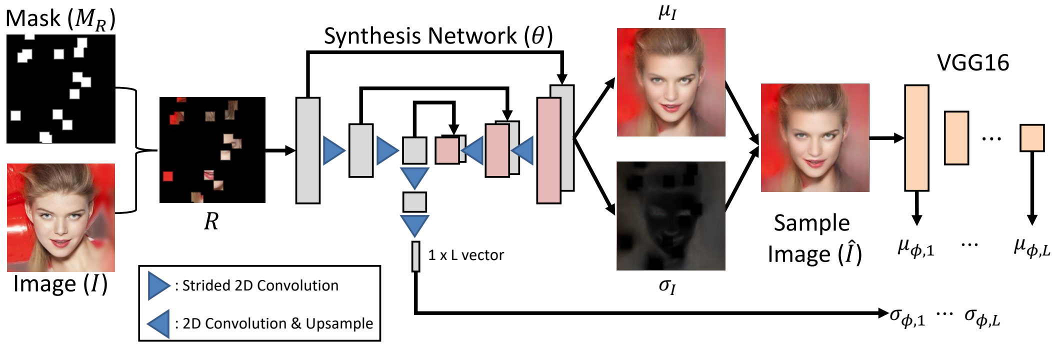
Mask (M_R)
Image (I)
R
Synthesis Network (θ)
1 x L vector
: Strided 2D Convolution
: 2D Convolution & Upsample
μ_I
σ_I
Sample Image ($\hat{I}$)
VGG16
$\mu_{\phi,1}$ ⋯ $\mu_{\phi,L}$
$\sigma_{\phi,1}$ ⋯ $\sigma_{\phi,L}$

Figure 2: **Image completion network.** Patch regions are specified by a binary mask M_R. The original image I is multiplied by M_R, and sent as input to the network, parameterized by θ. The outputs of the network are predictions for the average completed image, along with Gaussian standard deviation parameters σ_I and $\sigma_{\phi,l}, l \in \{1, \cdots L\}$, for the image and perceptual feature spaces.

not aim to maximize the information of the entire image from a few patches, and do not optimize a synthesis-based objective. Submodular functions have also been used for tracking and detecting objects [38, 52], and image collection summarization [43].

While this approach utilizes synthesis-based objective similar to image inpainting works, what differentiates this approach is that the primary focus is to retrieve the maximally informative patches instead of maximizing reconstruction quality. Therefore, the patch-selection method we use is network-agnostic and generalizes to different architectures. State-of-the-art architectures may provide crisper image reconstructions [32, 28, 7], but the UNet-style architecture we choose suffices in reconstruction quality and achieves the paper's main purpose of calculating mutual information.

3 Methods

We assume a training dataset $\mathcal{D}$ of images from a domain of interest and a fixed patch size s as input. Given an image $I \in \mathcal{R}^d$ from this domain at test time, we aim to construct an ordered set of k patches $R_k^* \in I$ carrying maximal information about I's visual content. Formally, we define R_k^* as the set of k patches with maximal pointwise mutual information (ρ) with I:

$$R_k^*(I) = \underset{R_k}{\operatorname{argmax}} \left(\rho(R_k, I)\right) = \underset{R_k}{\operatorname{argmax}} \left(\log \frac{p(I|R_k)}{p(I)}\right) = \underset{R_k}{\operatorname{argmax}} \left(\log p(I|R_k)\right). \tag{1}$$

We estimate $p(I|R_k)$ using a *synthesis-based approach* (see Figs. 1 and 2). We first specify a form for this distribution that captures both per-pixel and higher-order perceptual features. This formulation leads to a loss function with trainable parameters θ, which we implement with a CNN that completes images from a subset of visible patches.

Once trained, this network serves as a functional approximator to the pointwise mutual information, $\rho(\cdot,\cdot)$. Unfortunately, finding the optimal set R_k^* for a given k is a combinatorial, intractable problem. We instead leverage the fact that $\rho(\cdot,\cdot)$ is a monotone submodular function, and use a greedy patch selection strategy that is a provably good approximation to the optimal solution. Using a synthesis-driven objective to measure information content is attractive for several reasons. First, it is unsupervised, and therefore applicable to any image dataset. Second, since it is not tied to a particular task like classification or segmentation, our findings are applicable to any downstream task operating on the images. Finally, a synthesis network can tractably model long-range pixel dependencies, unlike approaches that only compare pixels over local neighborhoods [6, 19, 27].

3.1 Probabilistic Formulation

We build on existing works that combine per-pixel and "perceptual" losses for realistic image synthesis [22, 50]. A per-pixel loss treats each pixel as an independent observation, and the perceptual loss measures image differences in a more complex feature space $\phi(\cdot) : \mathcal{R}^d \rightarrow \mathcal{R}^{|\phi|}$ defined by the activations of an L-layer image classifier, e.g., VGG16 [39] ($L = 5$ in our experiments). We approximate $p(I|R)$ with these ideas using a factored form:

$$p(I|R;\theta,\phi) = p^{pixel}(I|R;\theta) \cdot p^{percep}(\phi(I)|R;\theta,\phi), \tag{2}$$

where θ are function parameters to be trained over data. We define the per-pixel distribution as:

$$p^{pixel}(I|R;\theta) = \mathcal{N}\left(I; \mu_I(R;\theta), \Sigma_I(R;\theta)\right), \tag{3}$$

where $\mu_I(\cdot;\theta) \in \mathcal{R}^d$ is a mean image, and $\Sigma_I(\cdot;\theta) \in \mathcal{R}^{d\times d}$ is a diagonal covariance matrix (i.e., effectively also in $\mathcal{R}^d$). Estimating a different variance for each pixel accounts for the heteroscedastic uncertainty of our task [24] due to certain pixels being easier to model than others given R.

We define the perceptual distribution as:

$$p^{percep}(\phi(I)|R;\theta,\phi) = \prod_{l=1}^{L} \mathcal{N}\left(\phi_l(I); \phi_l(\hat{I}), \sigma^2_{\phi,l}(R;\theta)\right), \tag{4}$$

where $\hat{I}$ is a sample from $p^{pixel}(\cdot|R;\theta)$, $\phi_l(\cdot)$ are features from layer l of the perceptual network, and $\sigma^2_{\phi,l}(\cdot;\theta) \in \mathcal{R}$ is a variance estimate for layer l.

3.2 Synthesis CNN and Loss Function

We use a UNet-style architecture [34] to implement the three functions $\mu_I(\cdot;\theta)$, $\Sigma_I(\cdot;\theta)$ and $\sigma^2_\phi(\cdot;\theta)$ (see Fig. 2). We output μ_I and Σ_I from the final convolutional layer of the UNet, at full image resolution. We estimate the vector σ^2_ϕ with a fully-connected layer applied to the activations of the middle (smallest) convolutional block of the UNet.

We represent R with a masked version of image I, obtained by multiplying a binary mask M_R element-wise with I. During training, we populate M_R with a random number of patches at random locations, to teach the model to complete arbitrarily-occluded images.

We optimize the network parameters by minimizing the negative log-likelihood of Eq. (2) over the training dataset $\mathcal{D}$:

$$\theta^* = \min_\theta \mathbb{E}_\mathcal{D}\left[-\log p^{pixel}(I|R;\theta) - \log p^{percep}(\phi(I)|R;\theta,\phi)\right]. \tag{5}$$

3.3 Patch Selection

At test time we use the trained network to calculate PMI between any set R and test image I. Constructing the optimal set R^*_k for value k requires a combinatorial search over many candidate regions with complexity $O(k^d)$, and is therefore intractable.

Fortunately, the PMI between a set of variables and any of its subsets is monotone and submodular in expectation (see supplementary material for proof). This means that the incremental information added by a patch about I decreases as the number of visible patches increases, and that if $R \in S \in I$, $PMI(R,I) \leq PMI(S,I)$. For such functions, a greedy selection of candidates is a provably good approximation, to within a factor of $1 - \frac{1}{e}$ of the optimal solution [26]. We perform the greedy search in a brute-force manner, for each patch over a fixed grid of image locations. Evaluating every pixel location is computationally intensive, so we set grid points to be a half-patch-size apart.

4 Experiments

We evaluate our work on three popular image datasets: CelebaHQ [23, 29] (high-res faces), STL-10 [9] (common animals/objects), and QuickDraw [16] (line drawings). We randomly split all of the CelebaHQ images to construct 28000/1000/1000 train/validation/test sets. We used a subset of the QuickDraw object classes, by taking the union of the 10 classes in the original study's experiments,

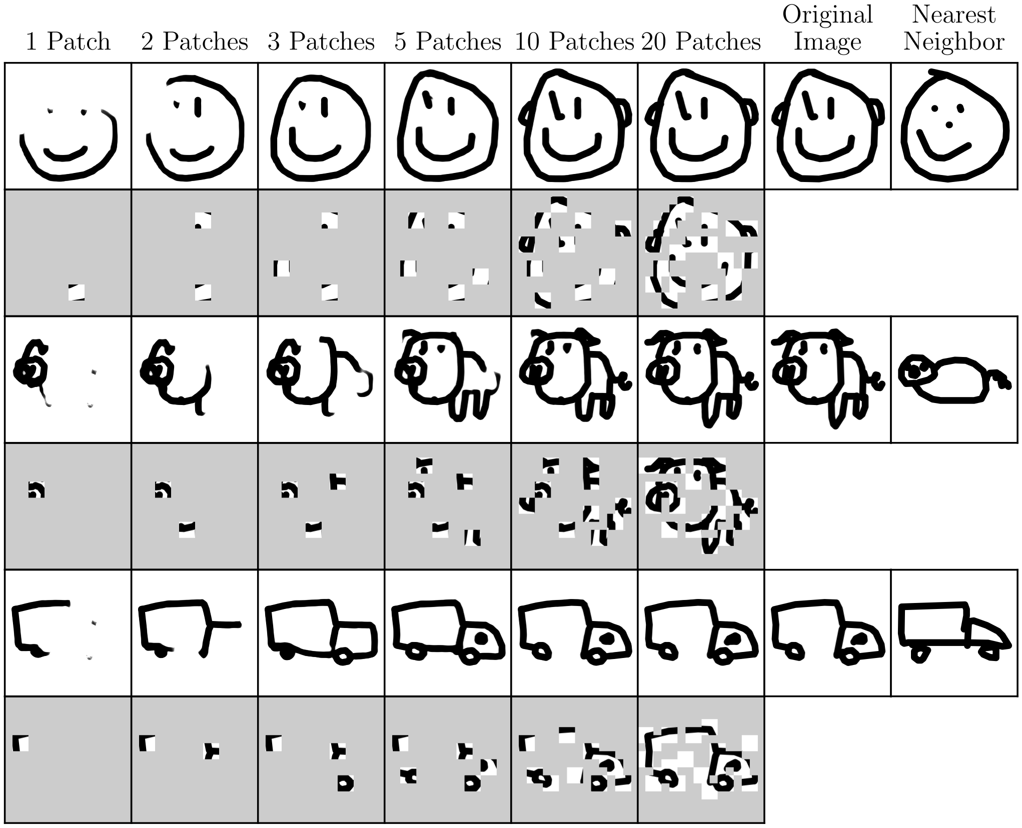

Figure 3: **QuickDraw reconstruction results for three test images.** Columns 1-6 of each odd row displays our method's outputs given different numbers of greedily-selected patches. The patches are shown in each even row, on top of a gray background for easier visualization. Ground truth images are shown in column 7. We also show the nearest neighbor from the training set in column 8.

and the object classes in STL-10, resulting in 17 classes[1]. We randomly sampled 5400/300/300 train/validation/test images per class. We found that STL-10 contains duplicate images between its unlabeled and labeled datasets, so we removed them from the unlabeled set, which we used for training. We used an 87,700/5000/1000 split of data.

We used constant patch areas of $1/64^{th}$ of the full image area of each dataset. For CelebAHQ and QuickDraw this resulted in 32×32 patches, and for STL-10 this resulted in 12×12 patches. We varied the number of patches from 1-25, covering roughly $1 - 40\%$ of the full image area. We trained separate models for each dataset, and our code is written in PyTorch.

4.1 Reconstruction Results

Figs. 3, 4 and 5 present sample reconstruction results for several test images from each dataset. We show both the reconstructed image, as well as the greedily selected patches from our method for different patch set sizes. We also display the nearest neighbor to the ground truth image in the training set, evaluated using the LPIPS perceptual error [50], for further context.

For QuickDraw, roughly 5 patches give close approximations of the ground truth for most cases, after which smaller details are added with subsequent patches. Selected patches tend to straddle areas with multiple semantic parts or corners. For example, in Fig. 3, the first patch selected for the face

[1] face, cat, pig, crab, rabbit, garden, owl, mosquito, yoga, airplane, bird, car, dog, horse, monkey, cruise ship, truck

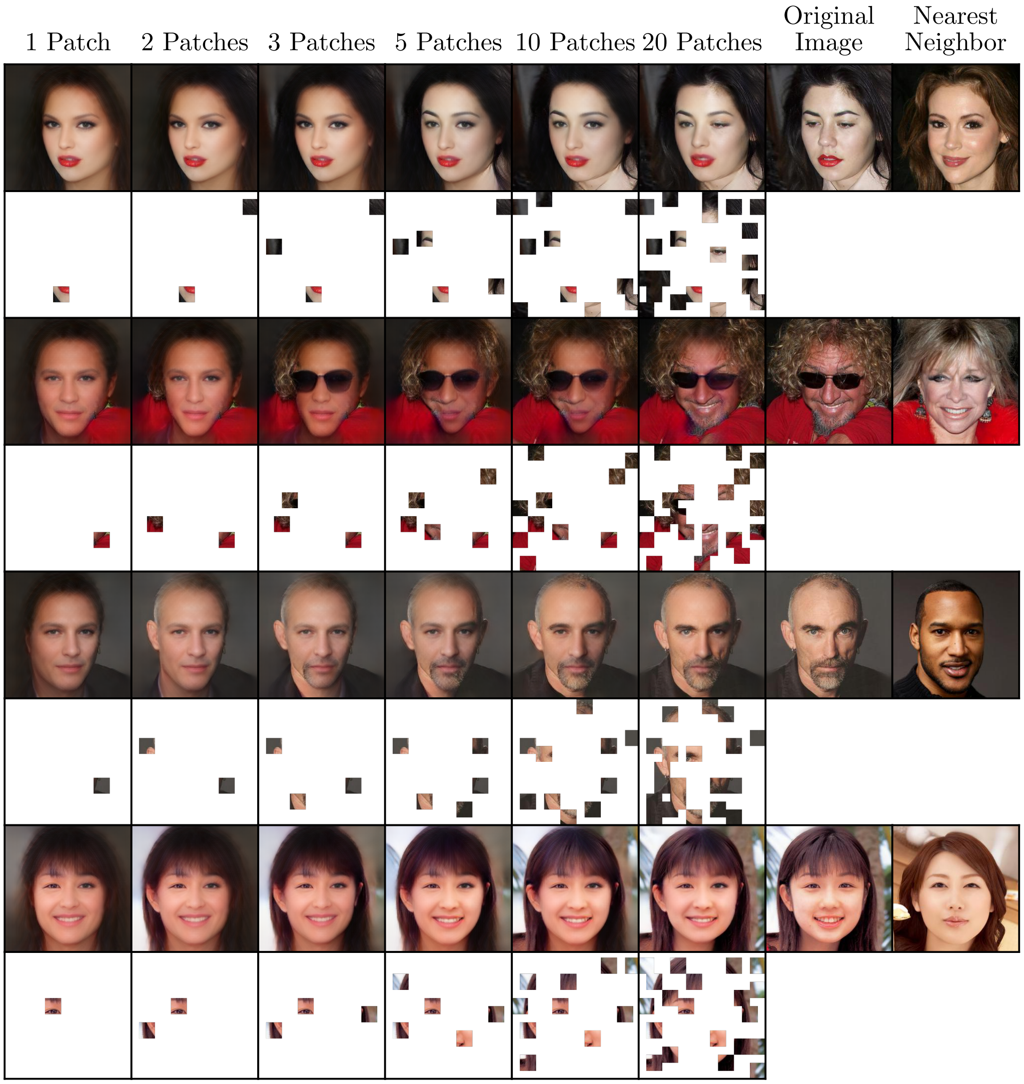

Figure 4: **CelebaHQ reconstruction results.** Column format is the same as in Fig. 3.

(example 1) straddles the chin and mouth, and the first patch for the truck (3rd example) is placed at its top-left corner.

CelebaHQ contains variabilities in facial attributes, poses, apparel, and background (see Fig. 4). Our synthesis method is usually able to establish gender with one greedily-selected patch, such as in examples 1,3, and 4. The first patch on the lip for example 1, for instance, helps establish both gender and pose. The fourth example is a case where patches are placed to help fill in both complex background details along with the face. The second example is particularly challenging due to the atypical hair, pose, and apparel combination.

STL-10 is the most challenging of the three datasets, consisting of more unconstrained images, with complex backgrounds. But even for this dataset, our method performs well, and is able to recover a surprising number of details within 10 patches. Some interesting landmarks captured by the patches include the ear of one dog (1st example, 2 patches) as well as the nose of a second dog in the same scene (1st example, 5 patches), part of the leg and beak of the bird (2nd example), and the center ridge of the boat (3rd example, 5 patches).

We numerically evaluated the reconstruction results against a baseline, random patch selection approach. The random selector places patches with equal probability at the same grid locations we

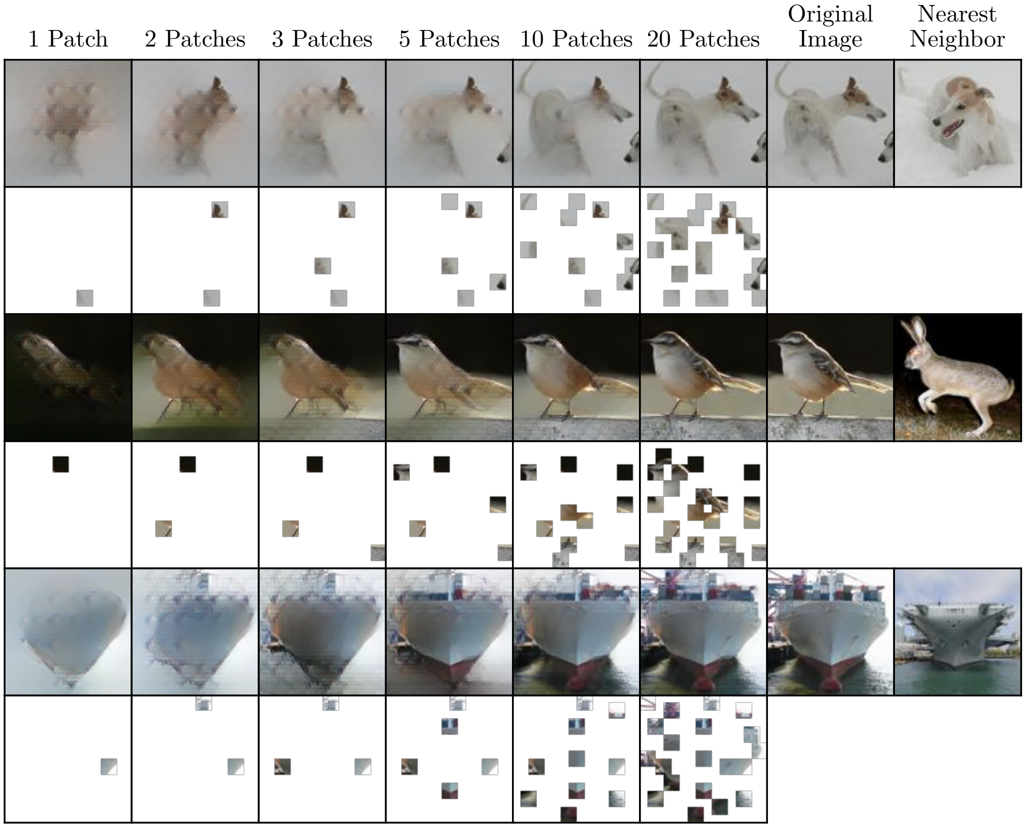

Figure 5: **STL-10 reconstruction results.** Column format is the same as in Fig. 3.

consider for the greedy search (see Sec. 3.3). For QuickDraw, we added a constraint that the patch must contain at least one black (foreground) pixel. Fig. 6 presents our results. Greedy selection outperforms random, with the difference more accentuated as the number of patches approaches 25. These results demonstrate that not all patches in an image are equal in regards to information content, and that careful selection is critical.

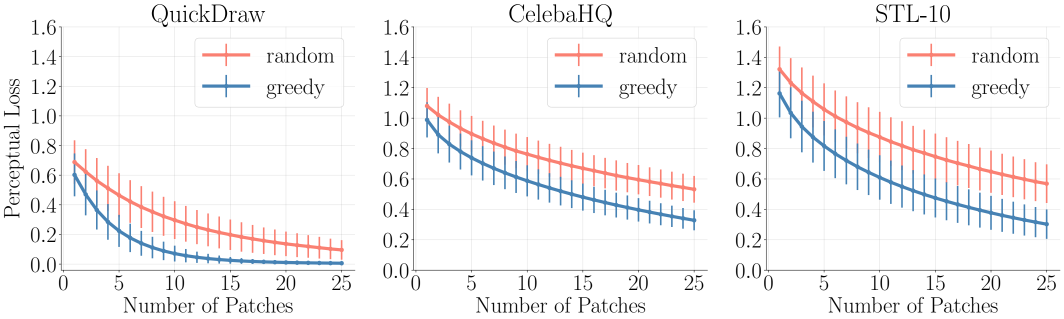

Figure 6: **Perceptual reconstruction error vs. number of input patches.** We randomly selected 1000 test images from each dataset and measured the LPIPS perceptual error [50] of our synthesis model's reconstructions given a varying number of visible input patches. We plot the mean error per set size, along with standard deviations in bars. Greedy patch selection significantly outperforms random selection, with the difference growing as $k \rightarrow 25$ for CelebaHQ and STL-10. This result shows that a sparse set of areas of the image are particularly informative.

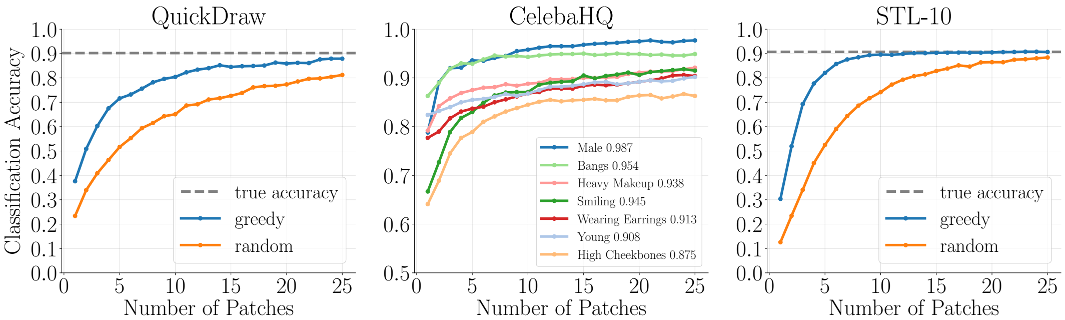

Figure 7: **Classification results.** We report accuracies of image classifiers on images synthesized by our approach from different numbers of greedily selected patches. For STL-10 and QuickDraw, we also show accuracies for random patch selection and the true accuracy of the classifier on images from that dataset. We omitted random results for CelebaHQ for the sake of figure clarity, and included the true classifier accuracy in the legend.

4.2 Patches for Classification

We explore the viability of using our patch sets for image classification. Instead of designing a new classifier that handles arbitrary patches as input, we evaluate a standard image classifier's accuracy on synthesized images produced from different sets of input patches. Each of the three datasets provide annotations: QuickDraw and STL-10 for object classes, and CelebaHQ for 40 facial features. We selected seven attributes for CelebaHQ capturing a broad span of face characteristics. We use ResNet-51 [17] architectures for each classifier. We use a multiclass softmax loss for QuickDraw (17-way) and STL-10 (10-way), and a binary cross-entropy loss for each CelebaHQ attribute.

Fig. 7 presents our method's accuracy versus patch set size. We show random patch selection as a baseline for QuickDraw and STL-10, and omit it for CelebaHQ to avoid overcrowding the figure (full results are in supplementary material). Greedy selection outperforms random for QuickDraw and STL-10 by a considerable margin, in line with our results for image reconstruction in Fig. 6. Greedy selection achieves within 5% of the true classifier accuracy at roughly 10-15 patches for QuickDraw, and 5-10 patches for STL-10.

Greedy selection performs well on CelebaHQ as well, though we do observe variability across attributes. For example, it struggles with few patches on *high cheekbones* and *smiling*, likely because slight variations to these features (e.g., a neutral vs. smiling mouth) are not particularly informative for reconstructing the full image.

5 Discussion and Conclusion

There are several avenues of further exploration given our results. It would be interesting to disentangle the information of a patch into its location and content components. For instance, the first QuickDraw example in Fig. 3 of a face shows how one carefully placed patch straddling the mouth and chin immediately constructs most of the face. Given that the image is from QuickDraw, clearly both the visual feature and its coordinate in image space say something about the object class. It is likely that our method uses such biases, which exist widely in vision datasets [41, 42], to its advantage.

Our experiments demonstrate that our minimal patch sets also contain appropriate information for classification tasks. Admittedly, more analysis may be required to apply those ideas in practice. First, it would be useful to show that the patches can be informative when given sequentially to an attention-based classifier to minimize computation. Second, our process for returning patches involves synthesizing a full image for each considered patch location, for all k iterations. This may be improved upon by using the per-pixel uncertainty map σ_I returned by our network to prune locations worth exploring. This would be more similar in spirit to ideas in sequential attention [12, 30].

Our results show that images from several popular datasets can be accurately inferred by just observing a few patches in sequential order, given prior data from those domains. This form of data redundancy has not been explored in depth by past work, and has natural ties to sequential attention processes of humans and recent vision algorithms, as well as image compression. At the core of our approach is a measure of pointwise mutual information captured by a synthesis model that does not rely on any supervised information. Quantifying information about a scene from a subset of patches specifies where an attention-based learner should look next in a task-agnostic way. In addition to image reconstruction, our experiments demonstrate these minimal patch sets contain appropriate information for potential downstream tasks like classification. In the case of low-bandwidth computing, this approach could process high-resolution images piecemeal and feed patches into the classifier.

The pointwise mutual information approach is both task-agnostic and network-agnostic. Other potential applications of this method include quantifying how much information a network can learn from seeing a face occluded by a face-mask and potentially detect which patches a face recognition system must focus on to authenticate a user. Furthermore, neural networks with attention-mechanisms, such as transformers, identify which parts of the image the model focuses on. Alternatively, a greedy search with among patches can also help us dissect where the network gains information to understand a scene. While this paper is not geared toward one single application, the method we propose demonstrates which information neural networks find most valuable, which can then be leveraged for various downstream tasks.

References

[1] Alan Allport. Visual attention. 1989.

[2] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6077–6086, 2018.

[3] Guy Ben-Yosef, Liav Assif, and Shimon Ullman. Full interpretation of minimal images. *Cognition*, 171:65–84, 2018.

[4] Ali Borji, Dicky N Sihite, and Laurent Itti. What/where to look next? modeling top-down visual attention in complex interactive environments. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(5):523–538, 2013.

[5] Philippe Boulinguez and Vincent Nougier. Control of goal-directed movements: the contribution of orienting of visual attention and motor preparation. *Acta Psychologica*, 103(1-2):21–45, 1999.

[6] Yuri Boykov and Gareth Funka-Lea. Graph cuts and efficient nd image segmentation. *International journal of computer vision*, 70(2):109–131, 2006.

[7] Weiwei Cai and Zhanguo Wei. Piigan: generative adversarial networks for pluralistic image inpainting. *IEEE Access*, 8:48451–48463, 2020.

[8] Kan Chen, Jiang Wang, Liang-Chieh Chen, Haoyuan Gao, Wei Xu, and Ram Nevatia. Abc-cnn: An attention based convolutional neural network for visual question answering. *arXiv preprint arXiv:1511.05960*, 2015.

[9] 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*, pages 215–223, 2011.

[10] John Colombo. The development of visual attention in infancy. *Annual review of psychology*, 52(1):337–367, 2001.

[11] Jia Deng, Jonathan Krause, and Li Fei-Fei. Fine-grained crowdsourcing for fine-grained recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2013.

[12] Burkhart Fischer and Heike Weber. Express saccades and visual attention. *Behavioral and Brain Sciences*, 16(3):553–567, 1993.

[13] Alexandra Frischen, Andrew P Bayliss, and Steven P Tipper. Gaze cueing of attention: visual attention, social cognition, and individual differences. *Psychological bulletin*, 133(4):694, 2007.

[14] Jianlong Fu, Heliang Zheng, and Tao Mei. Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4438–4446, 2017.

[15] Tom Gaens, Frederik Maes, Dirk Vandermeulen, and Paul Suetens. Non-rigid multimodal image registration using mutual information. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 1099–1106. Springer, 1998.

[16] David Ha and Douglas Eck. A neural representation of sketch drawings. In *ICLR 2018*, 2018.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[18] James E Hoffman. Visual attention and eye movements. *Attention*, 31:119–153, 1998.

[19] Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H Adelson. Crisp boundary detection using pointwise mutual information. In *European Conference on Computer Vision*, pages 799–814. Springer, 2014.

[20] Peng Jiang, Haibin Ling, Jingyi Yu, and Jingliang Peng. Salient region detection by ufo: Uniqueness, focusness and objectness. In *Proceedings of the IEEE international conference on computer vision*, pages 1976–1983, 2013.

[21] Zhuolin Jiang and Larry S Davis. Submodular salient region detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2043–2050, 2013.

[22] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.

[23] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[24] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in neural information processing systems*, pages 5574–5584, 2017.

[25] Gunhee Kim, Eric P Xing, Li Fei-Fei, and Takeo Kanade. Distributed cosegmentation via submodular optimization on anisotropic diffusion. In *2011 International Conference on Computer Vision*, pages 169–176. IEEE, 2011.

[26] Andreas Krause and Daniel Golovin. Submodular function maximization., 2014.

[27] Stan Z Li. *Markov random field modeling in computer vision*. Springer Science & Business Media, 2012.

[28] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 85–100, 2018.

[29] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[30] Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. Recurrent models of visual attention. In *Advances in neural information processing systems*, pages 2204–2212, 2014.

[31] Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. Dual attention networks for multimodal reasoning and matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 299–307, 2017.

[32] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016.

[33] Ronald A Rensink. The dynamic representation of scenes. *Visual cognition*, 7(1-3):17–42, 2000.

[34] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[35] Ruth Rosenholtz. Capabilities and limitations of peripheral vision. *Annual Review of Vision Science*, 2:437–457, 2016.

[36] Daniel Rueckert, Matthew J Clarkson, Derek LG Hill, and David John Hawkes. Non-rigid registration using higher-order mutual information. In *Medical Imaging 2000: Image Processing*, volume 3979, pages 438–447. International Society for Optics and Photonics, 2000.

[37] Daniel B Russakoff, Carlo Tomasi, Torsten Rohlfing, and Calvin R Maurer. Image similarity using mutual information of regions. In *European Conference on Computer Vision*, pages 596–607. Springer, 2004.

[38] Jianbing Shen, Zhiyuan Liang, Jianhong Liu, Hanqiu Sun, Ling Shao, and Dacheng Tao. Multiobject tracking by submodular optimization. *IEEE transactions on cybernetics*, 49(6):1990–2001, 2018.

[39] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[40] Anuj Srivastava, Ann B Lee, Eero P Simoncelli, and S-C Zhu. On advances in statistical modeling of natural images. *Journal of mathematical imaging and vision*, 18(1):17–33, 2003.

[41] Tatiana Tommasi, Novi Patricia, Barbara Caputo, and Tinne Tuytelaars. A deeper look at dataset bias. In *Domain adaptation in computer vision applications*, pages 37–55. Springer, 2017.

[42] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *CVPR 2011*, pages 1521–1528. IEEE, 2011.

[43] Sebastian Tschiatschek, Rishabh K Iyer, Haochen Wei, and Jeff A Bilmes. Learning mixtures of submodular functions for image collection summarization. In *Advances in neural information processing systems*, pages 1413–1421, 2014.

[44] Shimon Ullman, Liav Assif, Ethan Fetaya, and Daniel Harari. Atoms of recognition in human and computer vision. *Proceedings of the National Academy of Sciences*, 113(10):2744–2749, 2016.

[45] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[46] Paul Viola and William M Wells III. Alignment by maximization of mutual information. *International journal of computer vision*, 24(2):137–154, 1997.

[47] Luis Von Ahn, Ruoran Liu, and Manuel Blum. Peekaboom: a game for locating objects in images. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 55–64, 2006.

[48] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164, 2017.

[49] Tianjun Xiao, Yichong Xu, Kuiyuan Yang, Jiaxing Zhang, Yuxin Peng, and Zheng Zhang. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 842–850, 2015.

[50] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018.

[51] Heliang Zheng, Jianlong Fu, Tao Mei, and Jiebo Luo. Learning multi-attention convolutional neural network for fine-grained image recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 5209–5217, 2017.

[52] Fan Zhu, Zhuolin Jiang, and Ling Shao. Submodular object recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2457–2464, 2014.

Finding Maximally Informative Patches in Images

A $E_{R,I}[PMI(R,I)]$, where $R \subseteq I$, is submodular and monotonic.

Proof. $E_{R,I}[PMI(R,I)]$ is the mutual information between R and I, which we denote $MI(R,I)$. Rewriting in terms of informational entropy we obtain:

$$MI(R,I) = H(R) + H(R|I) = H(R), \tag{6}$$

since $H(R|I) = 0$ when $R \subseteq I$. Entropy is both submodular and monotonic. To see this, let $R \subseteq S \subseteq I$, and let $z \in I \setminus S$ be an unobserved patch. Then:

$$\begin{align} H(R,z) - H(R) &= H(z|R) \tag{7} \\ &\geq H(z|R \cup (S \setminus R)) \tag{8} \\ &= H(z|S) \tag{9} \\ &= H(S,z) - H(S), \tag{10} \end{align}$$

where step 8 follows from the fact that conditioning never increases entropy (“information never hurts”). Therefore, entropy is submodular. Furthermore, since adding a random variable to a set can only increase the set’s entropy, $H(R) \leq H(S)$, and entropy is also monotonic.

B Additional visual examples

See Figs. 8, 9, and 10 below for additional reconstruction results for each dataset.

C Celeba-HQ classification plots

See Fig. 11 for a breakdown of accuracies of our approach vs. random patch selection for each attribute.

D Visual examples ordered by difficulty

Figs. 12, 13 and 14 show test images ordered by the number of patches needed by our method to reconstruct them to a given accuracy threshold. This metric correlates well with the visual complexity of the images.

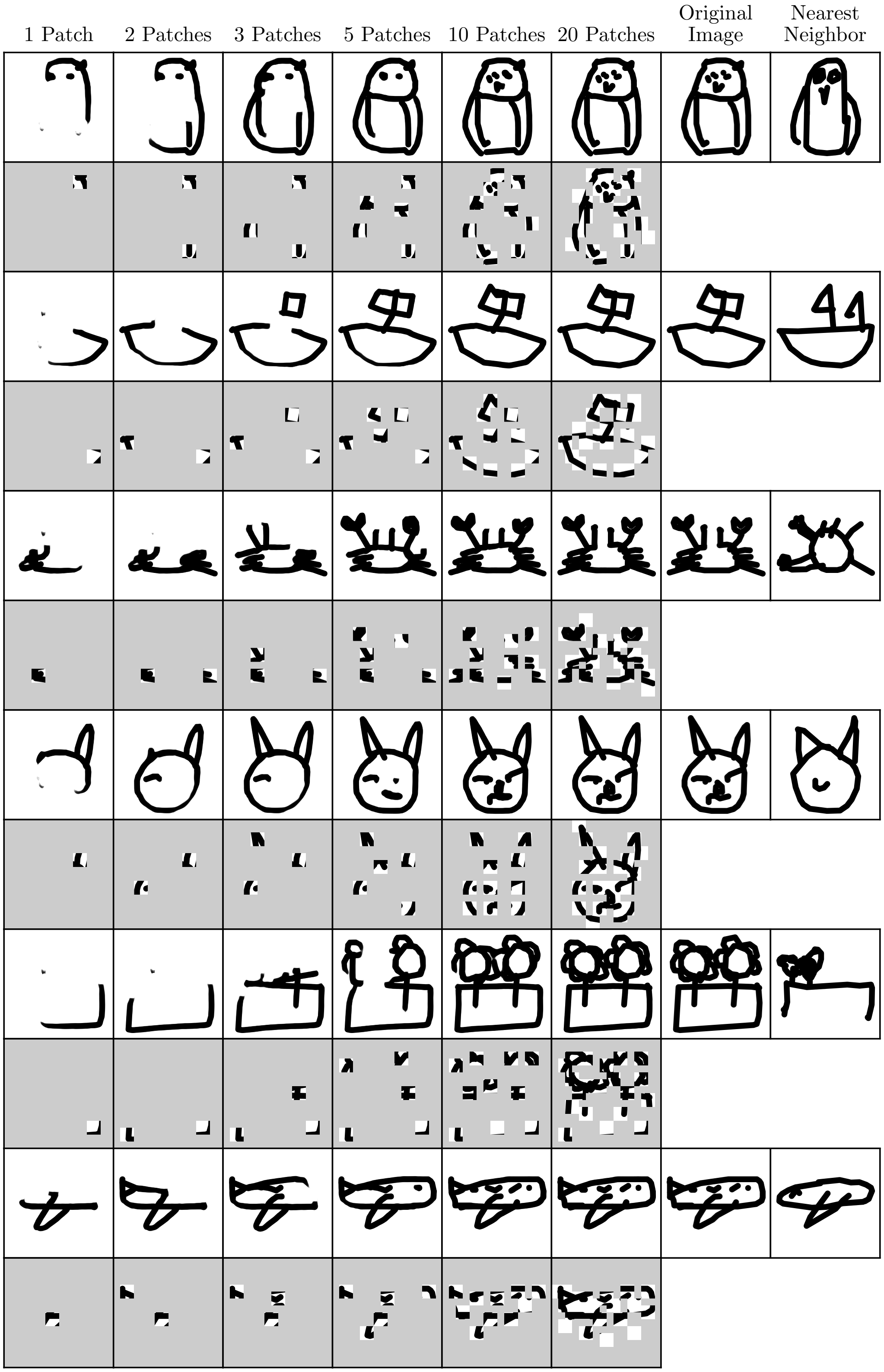

Figure 8: **Additional Quickdraw reconstruction results.** Column format is the same as in Fig. 3.

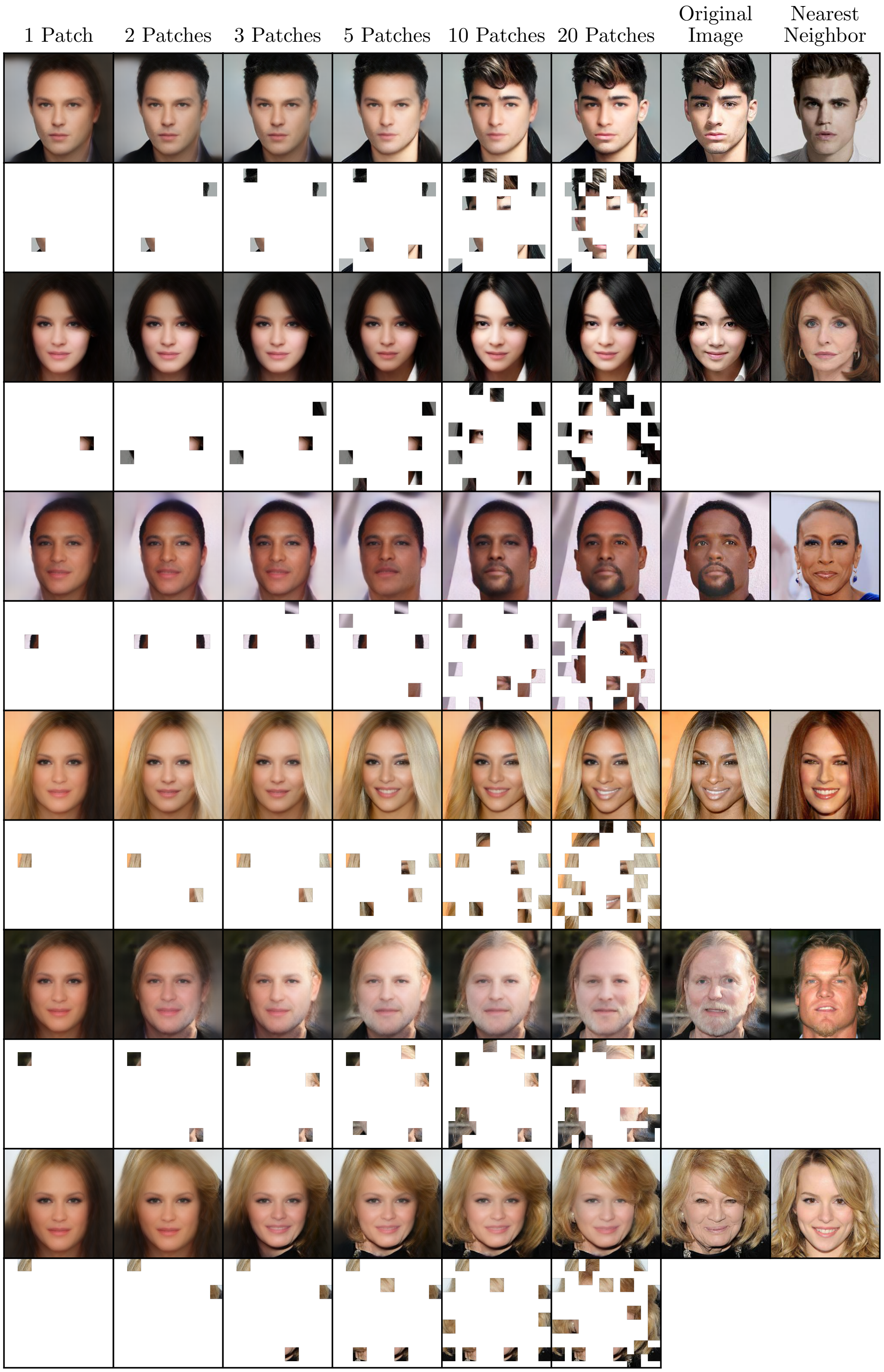

Figure 9: **Additional CelebaHQ reconstruction results.** Column format is the same as in Fig. 3.

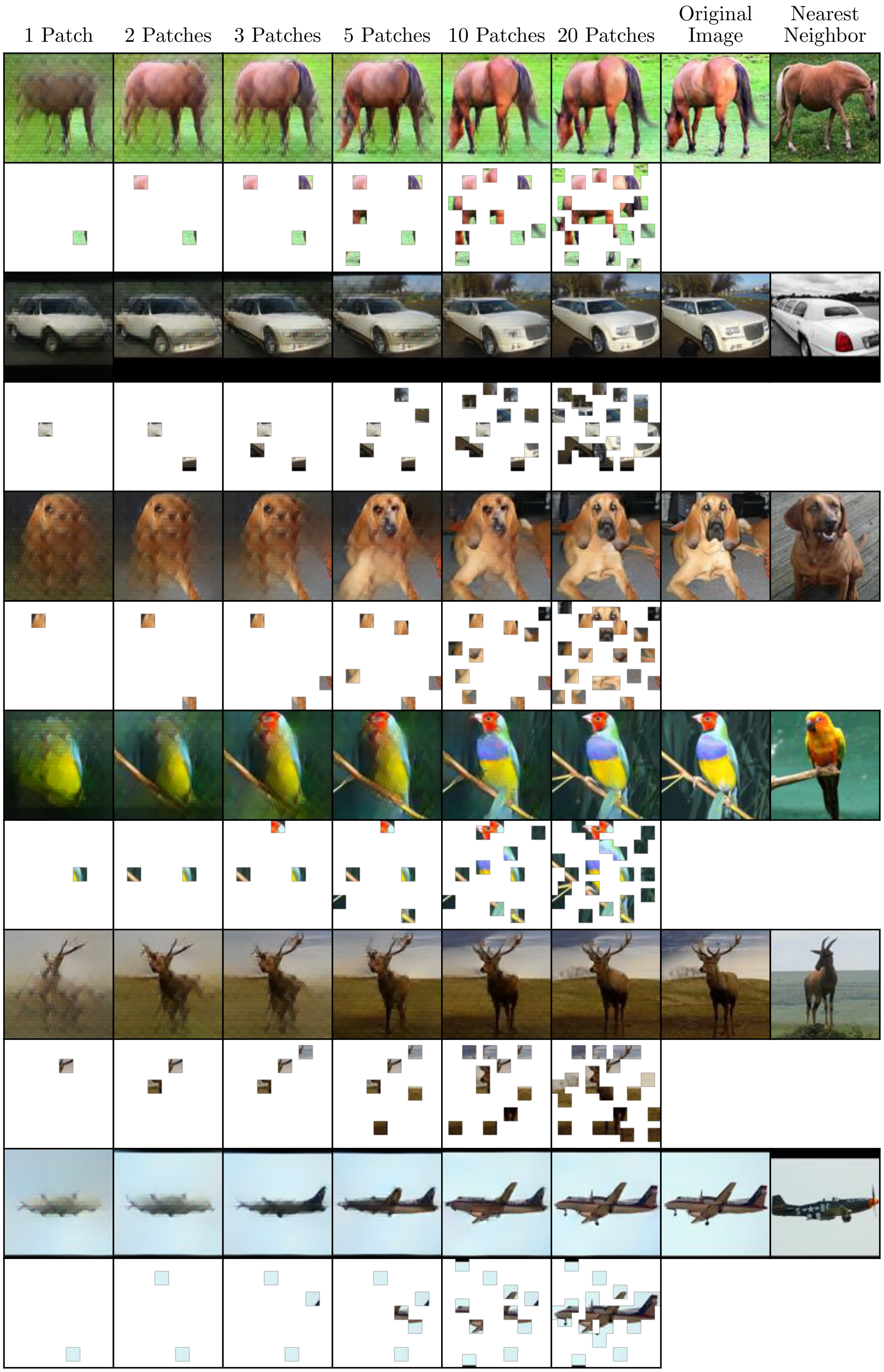

Figure 10: **Additional STL-10 reconstruction results.** Column format is the same as in Fig. 3.

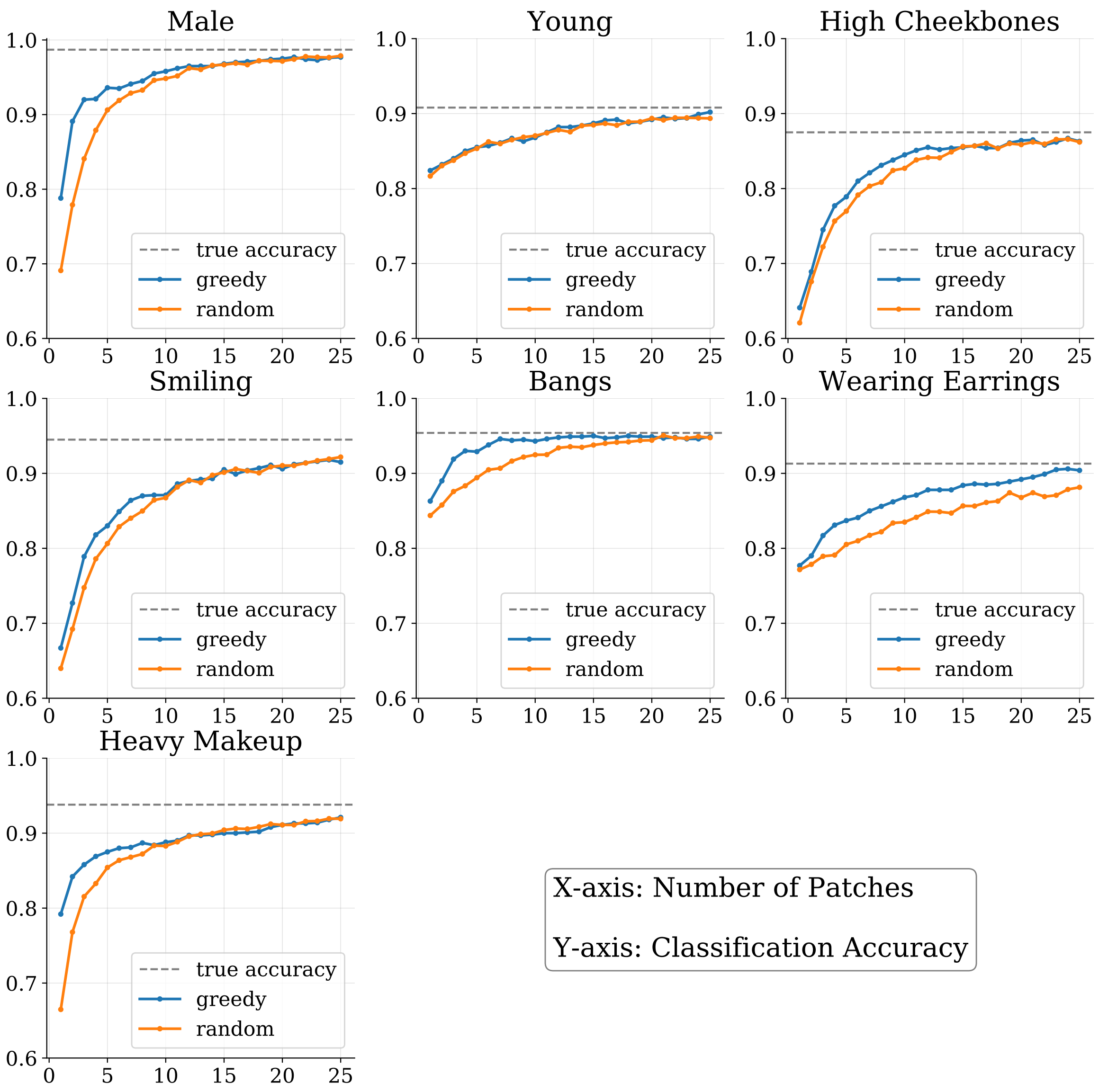

Figure 11: **Classification results for CelebaHQ.** A comparison of greedy vs. random patch selection for classification accuracy which we omitted from the main text (Fig. 7) due to space constraints. Greedy selection outperforms random selection in general, although the difference is smaller for *Young*, *High Cheekbones*, and *Smiling*.

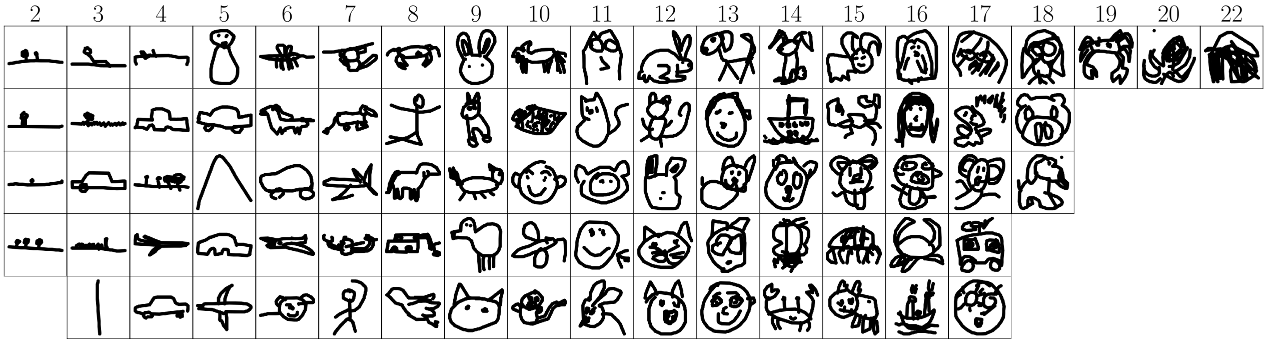

Figure 12: **QuickDraw images sorted by reconstruction difficulty.** We display randomly selected images in the QuickDraw test set sorted by the number of greedily-selected patches needed to achieve a negative log-probability (Eq. 5) below a threshold of 2.0. The number of patches correlates with image complexity.

r

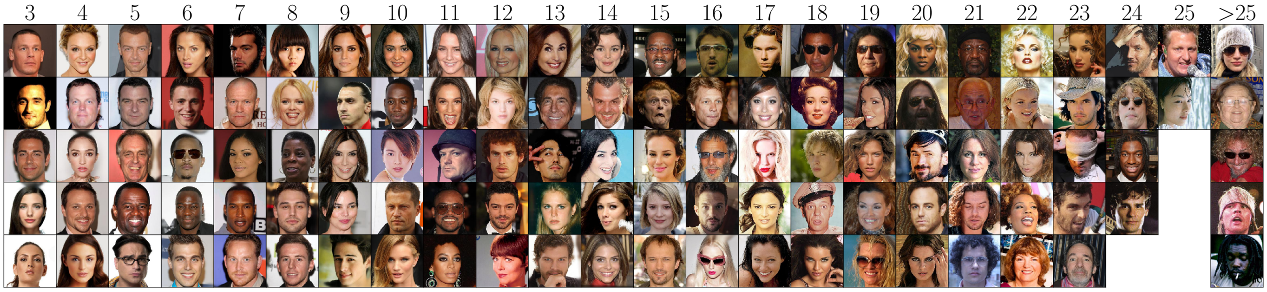

Figure 13: **CelebaHQ images sorted by reconstruction difficulty.** Column format is the same as Fig. 12. The negative log probability threshold is 5.5. Images with solid-colored backgrounds require fewer patches, and faces with non-standard image features and apparel require more patches.

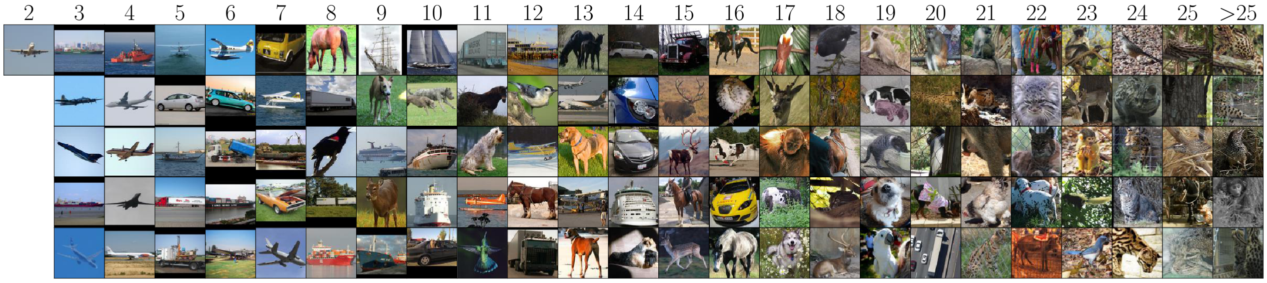

Figure 14: **STL-10 images sorted by reconstruction difficulty.** Column format is the same as Fig. 12. The negative log probability threshold is 5.5. Images depicting airplanes, trucks, and cars require fewer patches than those depicting animals. Animal images also tend to have more complex backgrounds.