IMPROVING THE ROBUSTNESS OF IMAGENET CLASSIFIERS USING ELEMENTS OF HUMAN VISUAL COGNITION

Anonymous authors
Paper under double-blind review

ABSTRACT

We investigate the robustness properties of image recognition models equipped with two features inspired by human vision, an explicit episodic memory and a shape bias, at the ImageNet scale. As reported in previous work, we show that an explicit episodic memory improves the robustness of image recognition models against small-norm adversarial perturbations under some threat models. It does not, however, improve the robustness against more natural, and typically larger, perturbations. Learning more robust features during training appears to be necessary for robustness in this second sense. We show that features derived from a model that was encouraged to learn global, shape-based representations (Geirhos et al., 2019) do not only improve the robustness against natural perturbations, but when used in conjunction with an episodic memory, they also provide additional robustness against adversarial perturbations. Finally, we address three important design choices for the episodic memory: memory size, dimensionality of the memories and the retrieval method. We show that to make the episodic memory more compact, it is preferable to reduce the number of memories by clustering them, instead of reducing their dimensionality.

1 INTRODUCTION

ImageNet-trained deep neural networks (DNNs) are state of the art models for a range of computer vision tasks and are currently also the best models of the human visual system and primate visual systems more generally (Schrimpf et al., 2018). Yet, they have serious deficiencies as models of human and primate visual systems: 1) they are extremely sensitive to small adversarial perturbations imperceptible to the human eye (Szegedy et al., 2013), 2) they are much more sensitive than humans to larger, more natural perturbations (Geirhos et al., 2018), 3) they rely heavily on local texture information in making their predictions, whereas humans rely much more on global shape information (Geirhos et al., 2019; Brendel & Bethge, 2019), 4) a fine-grained, image-by-image analysis suggests that images that ImageNet-trained DNNs find hard to recognize do not match well with the images that humans find hard to recognize (Rajalingham et al., 2018).

Here, we add a fifth under-appreciated deficiency: 5) human visual recognition has a strong episodic component lacking in DNNs. When we recognize a coffee mug, for instance, we do not just recognize it as *a* mug, but as *this particular* mug that we have seen before or as a novel mug that we have not seen before. This sense of familiarity/novelty comes automatically, involuntarily, even when we are not explicitly trying to judge the familiarity/novelty of an object we are seeing. More controlled psychological experiments also confirm this observation: humans have a phenomenally good long-term recognition memory with a massive capacity even in difficult one-shot settings (Standing, 1973; Brady et al., 2008). Standard deep vision models, on the other hand, cannot perform this kind of familiarity/novelty computation naturally or automatically, since this information is available to a trained model only indirectly and implicitly in its parameters.

What does it take to address these deficiencies and what are the potential benefits, if any, of doing so other than making the models more human-like in their behavior? In this paper, we address these questions. We show that a minimal model incorporating an explicit key-value based episodic memory does not only make it psychologically more realistic, but also reduces the sensitivity to

small adversarial perturbations. It does not, however, reduce the sensitivity to larger, more natural perturbations and it does not address the heavy local texture reliance issue. In the episodic memory, using features from DNNs that were trained to learn more global shape-based representations (Geirhos et al., 2019) addresses these remaining issues and moreover provides additional robustness against adversarial perturbations. Together, these results suggest that two basic ideas motivated and inspired by human vision, a strong episodic memory and a shape bias, can make image recognition models more robust to both natural and adversarial perturbations at the ImageNet scale.

2 Related work

In this section, we review previous work most closely related to ours and summarize our own contributions.

To our knowledge, the idea of using an episodic cache memory to improve the adversarial robustness of image classifiers was first proposed in Zhao & Cho (2018) and in Papernot & McDaniel (2018). Zhao & Cho (2018) considered a differentiable memory that was trained end-to-end with the rest of the model. This makes their model computationally much more expensive than the cache models considered here, where the cache uses pre-trained features instead. The deep k-nearest neighbor model in Papernot & McDaniel (2018) and the "CacheOnly" model described in Orhan (2018) are closer to our cache models in this respect, however these works did not consider models at the ImageNet scale. More recently, Dubey et al. (2019) did consider cache models at the ImageNet scale (and beyond) and demonstrated substantial improvements in adversarial robustness under certain threat models.

None of these earlier papers addressed the important problem of robustness to natural perturbations and they did not investigate the effects of various cache design choices, such as the retrieval method (i.e. a continuous cache vs. nearest neighbor retrieval), cache size, dimensionality of the keys or the feature type used (e.g. texture-based vs. shape-based features), on the robustness properties of the cache model.

A different line of recent work addressed the question of robustness to natural perturbations in ImageNet-trained DNNs. In well-controlled psychophysical experiments with human subjects, Geirhos et al. (2018) compared the sensitivity of humans and ImageNet-trained DNNs to several different types of natural distortions and perturbations, such as changes in contrast, color or spatial frequency content of images, image rotations etc. They found that ImageNet-trained DNNs are much more sensitive to such perturbations than human subjects. More recently, Hendrycks & Dietterich (2019) introduced the ImageNet-C and ImageNet-P benchmarks to measure the robustness of neural networks against some common perturbations and corruptions that are likely to occur in the real world. We use the ImageNet-C benchmark below to measure the robustness of different models against natural perturbations.

This second line of work, however, did not address the question of adversarial robustness. An adequate model of the human visual system should be robust to both natural and adversarial perturbations.[1] Moreover, both properties are clearly desirable properties in practical image recognition systems, independent of their value in building more adequate models of the human visual system.

Our main contributions in this paper are as follows: 1) as reported in previous work (Zhao & Cho, 2018; Papernot & McDaniel, 2018; Orhan, 2018; Dubey et al., 2019), we show that an explicit cache memory improves the adversarial robustness of image recognition models at the ImageNet scale, but only under certain threat scenarios; 2) we investigate the effects of various design choices for the cache memory, such as the retrieval method, cache size, dimensionality of the keys and the feature type used for extracting the keys; 3) we show that caching, by itself, does not improve the robustness of classifiers against natural perturbations; 4) using more global, shape-based features (Geirhos et al., 2019) in the cache does not only improve robustness against natural perturbations, but also provides extra robustness against adversarial perturbations as well.[2]

[1] Two recent papers (Elsayed et al., 2018; Zhou & Firestone, 2019) suggested that humans might be vulnerable, or at least sensitive, to adversarial perturbations too. However, these results apply only in very limited experimental settings (e.g. very short viewing times in Elsayed et al. (2018)) and require relatively large and transferable perturbations, which often tend to yield meaningful features resembling the target class.

[2] All code and simulation results will be made available at: `https://github.com/`

3 METHODS

3.1 MODELS

Throughout the paper, we use pre-trained ResNet-50 models either on their own or as feature extractors (or "backbones") to build cache models that incorporate an explicit episodic memory storing low-dimensional embeddings (or keys) for all images seen during training (Orhan, 2018). The cache models in this paper are essentially identical to the "CacheOnly" models described in Orhan (2018). A schematic diagram of a cache model is shown in Figure 1.

We used one of the higher layers of a pre-trained ResNet-50 model as an embedding layer. Let $\phi(\mathbf{x})$ denote the d-dimensional embedding of an image $\mathbf{x}$ into this layer. The cache is then a key-value dictionary consisting of the keys $\mu_k \equiv \phi(\mathbf{x}_k)$ for each training image $\mathbf{x}_k$ in the dataset and the values are the corresponding class labels represented as one-hot vectors v_k. We normalized all keys to have unit l_2-norm.

When a new test image $\mathbf{x}$ is presented, the similarity between its key and all other keys in the cache is computed through:

$$\sigma_k(\mathbf{x}) \propto \exp(\theta \phi(\mathbf{x})^\top \mu_k) \tag{1}$$

and a distribution over labels is obtained by taking a weighted average of the values stored in the cache:

$$p_{\text{cache}}(\mathbf{y}|\mathbf{x}) = \frac{\sum_{k=1}^{K} v_k \sigma_k(\mathbf{x})}{\sum_{k=1}^{K} \sigma_k(\mathbf{x})} \tag{2}$$

where K denotes the number of items stored in the cache. The hyper-parameter θ in Equation 1 controls the sharpness of this distribution, with larger θ values producing sharper distributions. We optimized θ only in one of the experimental conditions below (the gray-box adversarial setting) by searching over 9 uniformly spaced values between 10 and 90 and fixed its value for all other conditions.

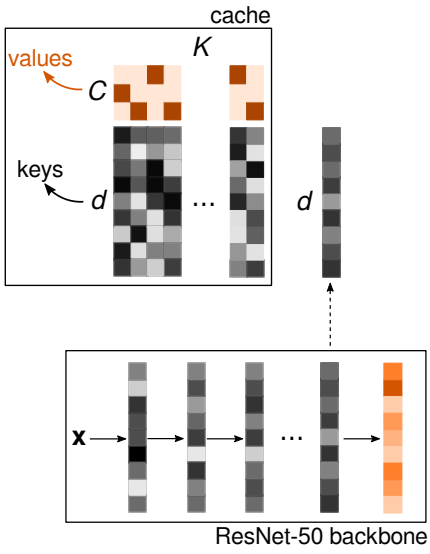

Figure 1: Schematic illustration of the cache model. The key for a new image $\mathbf{x}$ is compared with the keys in the cache. A prediction is made by a linear combination of the values weighted by the similarity to the corresponding keys.

Because we take all items in the cache into account in Equation 2, weighted by their similarity to the test item, we call this type of cache a *continuous cache* (Grave et al., 2016). An alternative (and more scalable) approach would be to perform a nearest neighbor search in the cache and consider only the most similar items in making predictions (Grave et al., 2017; Dubey et al., 2019). We compare the relative performance of these two approaches below.

For the embedding layer, we considered three choices (in descending order and using the layer names from the `torchvision.models` implementation of ResNet-50): `fc`, `avgpool`, and `layer4_bottleneck1_relu`. `fc` corresponds to the final softmax layer (we used the post-nonlinearity probabilities, not the logits), `avgpool` corresponds to the global average pooling layer right before the final layer and `layer4_bottleneck1_relu` is the output of the penultimate bottleneck block of the network. We also explored the use of lower layers as embeddings; however, these layers led to substantially worse clean and adversarial accuracies, hence they were not considered further. `layer4_bottleneck1_relu` is a $7 \times 7 \times 2048$-dimensional spatial layer; we applied a global spatial average pooling operation to this layer to reduce its dimensionality. This gave rise to $d = 1000$ dimensional keys for `fc` and $d = 2048$ dimensional keys for the other two layers.

To investigate the effect of different feature types on the robustness of the models, we also considered a ResNet-50 model jointly trained on ImageNet and Stylized-ImageNet datasets and then fine-tuned on ImageNet (Geirhos et al., 2019) (we used the pre-trained model provided by the authors). Following Geirhos et al. (2019), we call this model Shape-ResNet-50. Geirhos et al. (2019) argue that Shape-ResNet-50 learns more global, shape-based representations than a standard ImageNet-trained ResNet-50 (which instead relies more heavily on local texture) and produces predictions more in line with human judgments in texture vs. shape cue conflict experiments.

All experiments were conducted on the ImageNet dataset containing approximately 1.28M training images from 1000 classes and 50K validation images (Russakovsky et al., 2015). We note that using the full cache (i.e. a continuous cache) was computationally feasible in our experiments at the ImageNet scale. The largest cache we used (of size $1.28\text{M} \times 2048$) takes up $\sim 10.5\text{GB}$ of disk space when stored as a single-precision floating-point array.

3.2 PERTURBATIONS

Ideally, we want our image recognition models to be robust against both adversarial perturbations and more natural perturbations. This subsection describes the details of the natural and adversarial perturbations considered in this paper.

3.2.1 ADVERSARIAL PERTURBATIONS

Our experiments on adversarial perturbations closely followed the experimental settings described in Dubey et al. (2019). In particular, we considered three different threat models: white-box attacks, gray-box attacks, and black-box attacks.

White-box attacks: This is the strongest attack scenario. In this scenario, the attacker has full knowledge of the backbone model and the items stored in the cache.

Gray-box attacks: In this scenario, the attacker has full knowledge of the backbone model, but does not have access to the items stored in the cache. In many cases, this threat scenario is more realistic than the white-box or black-box settings, since the models used as feature extractors are usually publicly available (e.g. pre-trained ImageNet models), but the database of items stored using those features is private. In practice, for the cache models, we implemented the gray-box scenario by first running white-box attacks against the backbone model and then testing the resulting adversarial examples on the cache model.

Black-box attacks: This is the weakest attack scenario where the attacker does not know the backbone model or the items stored in the cache. For the cache models, we implemented the black-box scenario by running white-box attacks against a model different from the model used as the backbone and testing the resulting adversarial examples on the cache model as well as on the backbone itself. In practice, we used an ImageNet-trained ResNet-18 model to generate adversarial examples in this setting (note that we always use a ResNet-50 backbone in our models).

We chose a strong, state-of-the-art, gradient-based attack method called projected gradient descent (PGD) with random starts (Madry et al., 2017) to generate adversarial examples in all three settings. We used the Foolbox implementation of this attack (Rauber et al., 2017), `RandomStartProjectedGradientDescentAttack`, with the following attack parameters: `binary_search=False`, `stepsize=2/225`, `iterations=10`, `random_start=True`. We also controlled the total size of the adversarial perturbation as measured by the l_∞-norm of the perturbation normalized by the l_∞-norm of the clean image $\mathbf{x}$: $\epsilon \equiv ||\mathbf{x}_{\text{adv}} - \mathbf{x}||_\infty / ||\mathbf{x}||_\infty$. We considered six different ϵ values: 0.01, 0.02, 0.04, 0.06, 0.08, 0.1. In general, the attacks are expected to be more successful for larger ϵ values.

As recommended by Athalye et al. (2018), we used targeted attacks, where for each validation image we first chose a target class label different from the correct class label for the image and then ran the attack to return an image that was misclassified as belonging to the target class. In cases where the attack was not successful, the original clean image was returned, therefore the model had the same baseline accuracy on such failure cases as on clean images. We ran attacks starting from all validation images, hence the reported accuracies are averages over all validation images.

3.2.2 NATURAL PERTURBATIONS

To measure the robustness of image recognition models against natural perturbations, we used the recently introduced ImageNet-C benchmark (Hendrycks & Dietterich, 2019). ImageNet-C contains 15 different natural perturbations applied to each image in the ImageNet validation set at 5 different severity levels, for a total of $15 \times 5 \times 50\text{K} = 3.75\text{M}$ images. The perturbations in ImageNet-C come in four different categories: 1) *noise* perturbations (Gaussian, shot, and impulse noise), 2) *blur* perturbations (defocus, glass, motion, and zoom blur), 3) *weather* perturbations (snow, frost, fog, and

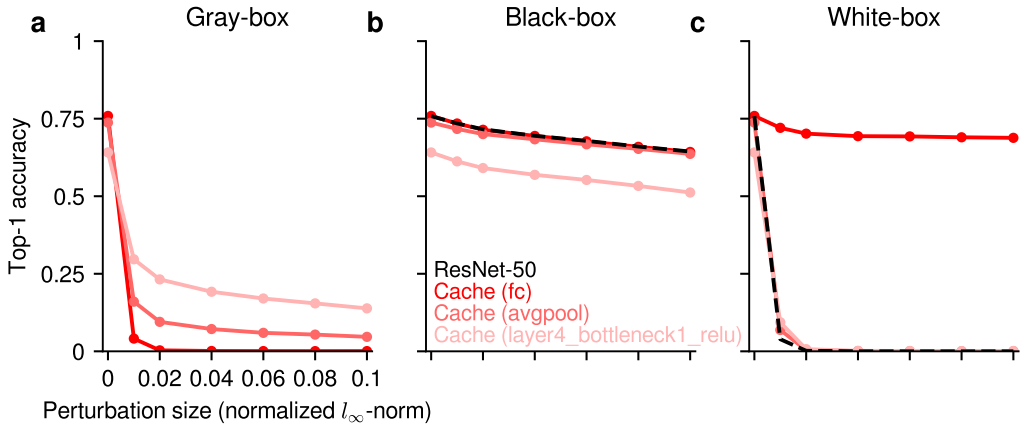

Figure 2: Top-1 accuracy of the ResNet-50 backbone and cache models in the (**a**) gray-box, (**b**) black-box and (**c**) white-box adversarial settings. The 0 perturbation size corresponds to the clean images. Note that the gray-box setting is meaningful for the cache models only and is not well-defined for the backbone ResNet-50 model.

brightness), and 4) *digital* perturbations (contrast, elasticity, pixelation, and JPEG compression). We refer the reader to Hendrycks & Dietterich (2019) for further details about the dataset.

To measure the robustness of a model against the perturbations in ImageNet-C, we use the mCE (mean corruption error) measure (Hendrycks & Dietterich, 2019). A model's mCE is calculated as follows. For each perturbation c, we first average the model's classification error over the 5 different severity levels s and divide the result by the average error of a reference classifier (which is taken to be the AlexNet): $CE_c \equiv \langle E_{s,c} \rangle_s / \langle E_{s,c}^{\text{AlexNet}} \rangle_s$. The overall performance on ImageNet-C is then measured by the mean CE_c averaged over the 15 different perturbation types c: $mCE \equiv \langle CE_c \rangle_c$. Dividing by the performance of a reference model in calculating CE_c ensures that different perturbations have roughly similar sized contributions to the overall measure *mCE*. Note that smaller *mCE* values indicate more robust classifiers.

4 Results

4.1 Caching improves robustness against adversarial perturbations

Figure 2 shows the adversarial accuracy in the gray-box, black-box, and white-box settings for cache models using different layers as embeddings. In the gray-box setting, lower layers showed more robustness at the expense of a reduction in clean accuracy, with the `layer4_bottleneck1_relu` layer achieving the highest gray-box accuracies.

In the black-box setting, we found that even large perturbation adversarial examples for the ResNet-18 model were not effective adversarial examples for the backbone ResNet-50 model (dashed line) or for the cache models, hence the models largely maintained their performance on clean images with a slight general decrease in accuracy for larger perturbation sizes.

In the white-box setting, we observed a divergence in behavior between `fc` and the other layers. The PGD attack was generally unsuccessful against the `fc` layer cache model, whereas for the other layers it was highly successful even for small perturbation sizes. The softmax non-linearity in `fc` was crucial for this effect, as it was substantially easier to run successful white-box attacks when the logits were used as keys instead. We thus attribute this effect to gradient obfuscation in the `fc` layer cache model (Athalye et al., 2018), rather than consider it as a real sign of adversarial robustness. Indeed, the gray-box adversarial examples (generated from the backbone ResNet-50 model) were very effective against the `fc` layer cache model (Figure 2a).

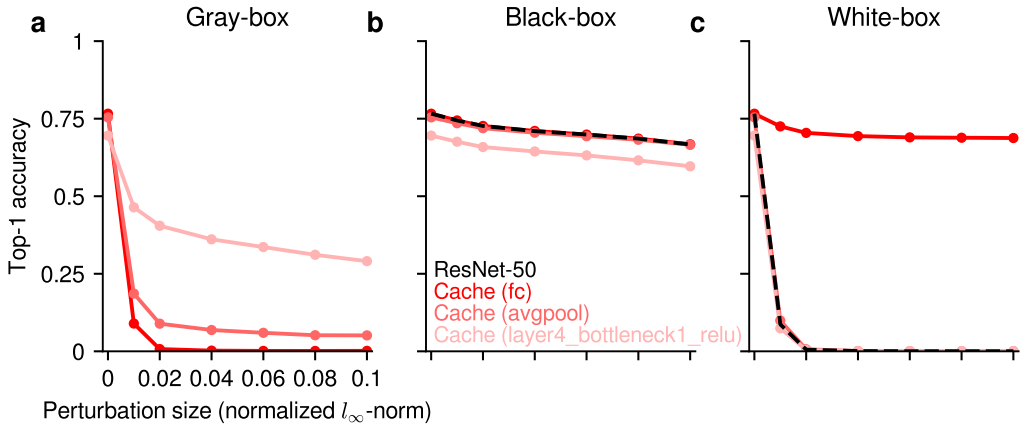

Figure 3: Similar to Figure 2, but with Shape-ResNet-50 as the backbone.

Table 1: Clean and adversarial accuracies of texture- and shape-based ResNet-50 backbones and cache models. The adversarial accuracies report the results for a standard normalized perturbation size of $\epsilon = 0.06$.

Model	Clean	Gray-box	Black-box	White-box
ResNet-50	0.758	–	0.678	0.000
Cache (`layer4_bottleneck1_relu`, texture)	0.641	0.170	0.552	0.001
Shape-ResNet-50	0.766	–	0.699	0.000
Cache (`layer4_bottleneck1_relu`, shape)	0.695	0.336	0.632	0.001

Qualitatively similar results were observed when Shape-ResNet-50 was used as the backbone instead of ResNet-50 (Figure 3). Table 1 reports the clean and adversarial accuracies for a subset of the conditions.

4.2 Cache design choices

In this subsection, we consider the effect of three cache design choices on the clean and adversarial accuracy of cache models: the size and dimensionality of the cache and the retrieval method.

Dubey et al. (2019) recently investigated the adversarial robustness of cache models with very large databases (databases of up to $K = 50$B items). Scaling up the cache model to very large databases requires making the cache memory as compact as possible and using a fast approximate nearest neighbor algorithm for retrieval from the cache (instead of using a continuous cache). There are at least two different ways of making the cache more compact: one can either reduce the number of items in the cache by clustering them, or alternatively one can reduce the dimensionality of the keys.

Dubey et al. (2019) made the keys more compact by reducing the original 2048-dimensional embeddings to 256 dimensions (an 8-fold compression) with online PCA and used a fast 50-nearest neighbor (50-nn) method for retrieval.

In our experiments, replacing the continuous cache with a 50-nn retrieval method did not have an adverse effect on adversarial and clean accuracies (Figure 4 and Table 2). This suggests that the continuous cache can be safely replaced with an efficient nearest neighbor algorithm to scale up the cache size without much effect on the model accuracy.

On the other hand, reducing the dimensionality of the keys from 2048 to 256 using online PCA over the training data resulted in a substantial drop in both clean and adversarial accuracies (Figure 4 and Table 2). Even a 4-fold reduction to 512 dimensions resulted in a large drop in accuracy. This implies that the higher layers of the backbone used for caching are not very compressible and drastic dimensionality reduction measures should be avoided to prevent a substantial decrease in accuracy.

Table 2: Clean and gray-box adversarial accuracies of different cache models. As in Figure 4, only the results for the `layer4_bottleneck1_relu` layer are shown. Colors highlight the retrieval method (continuous or 50-nn), cache dimensionality (full, 4- or 8-times reduced), and cache size (full, 4- or 8-times reduced).

Model	Clean	Gray-box ($\epsilon = 0.06$)
Cache (continuous, full-dims., full-cache)	0.641	0.170
Cache (50-nn, full-dims., full-cache)	0.656	0.181
Cache (50-nn, full-dims., 1/4-cache)	0.626	0.151
Cache (50-nn, full-dims., 1/8-cache)	0.612	0.143
Cache (50-nn, 1/4-dims., full-cache)	0.516	0.109
Cache (50-nn, 1/8-dims., full-cache)	0.498	0.103

Reducing the cache size by the same amount (4-fold or 8-fold compression) by clustering the items in the cache with a mini-batch k-means algorithm resulted in a significantly smaller decrease in accuracy (Figure 4 and Table 2): for example, an 8-fold reduction in dimensionality led to a clean accuracy of 49.8%, whereas an 8-fold reduction in the cache size instead resulted in a clean accuracy of 61.2%. This suggests that the cluster structure in the keys is much more prominent than the linear correlation between the dimensions. Therefore, to make the cache more compact, given a choice between reducing the dimensionality vs. reducing the number of items by the same amount, it is preferable to choose the second option for better accuracy.

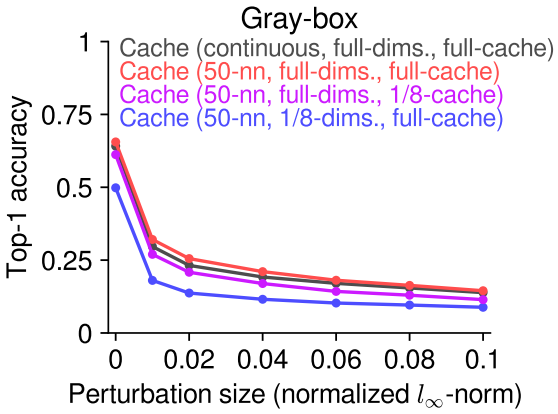

Figure 4: The effects of three cache design choices on the clean and adversarial accuracy in the gray-box setting. The results shown here are for the `layer4_bottleneck1_relu` layer. Similar results were observed for other layers.

4.3 Caching does not improve robustness against natural perturbations

We have seen that caching can improve robustness against gray-box adversarial perturbations. Does it also improve robustness against more natural perturbations? Table 3 shows that the answer is no. On ImageNet-C, the backbone ResNet-50 model yields an mCE of 0.764. The best cache model obtained approximately the same mCE score. We suggest that this is because caching improves robustness only against small-norm perturbations, whereas natural perturbations in ImageNet-C are typically much larger. Even the smallest size perturbations in ImageNet-C are clearly visible to the eye (Hendrycks & Dietterich, 2019) and we calculated that even these smallest size perturbations have an average normalized l_∞-norm of $\epsilon \approx 1$ for all perturbation types, compared to the largest adversarial perturbation size of $\epsilon = 0.1$ considered in this paper. This result is also consistent with a similar observation made by Gu et al. (2019) suggesting that perturbations occurring between neighboring frames in natural videos are much larger in magnitude than adversarial perturbations. We conjecture that robustness against such large perturbations cannot be achieved with test-time only interventions such as caching and requires learning more robust backbone features in the first place.

4.4 Using shape-based features in the cache improves both adversarial and natural robustness

To investigate the effect of different kinds of features in the cache, we repeated our experiments using cache models with Shape-ResNet-50 as the backbone (see *Methods* for further details about Shape-

Table 3: ImageNet-C results. The numbers indicate corruption errors (CE) for specific corruption types and the mean CE scores as percentages. More robust models correspond to smaller numbers. For the cache models, we only show the results for the best models (the `fc` cache model in both cases). Colors represent noise, blur, weather and digital perturbations.

Model	mCE	Gauss	Shot	Impul.	Defoc.	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contr.	Elastic	Pixel	JPEG
ResNet-50	76.4	78	80	80	75	89	78	80	78	75	67	57	72	86	77	76
Cache (texture)	76.4	78	79	80	75	89	78	80	78	75	67	57	72	86	77	76
Shape-ResNet-50	73.5	74	75	75	72	86	74	80	75	73	67	55	68	81	75	72
Cache (shape)	73.5	74	75	75	72	86	74	80	75	73	67	55	68	81	75	72

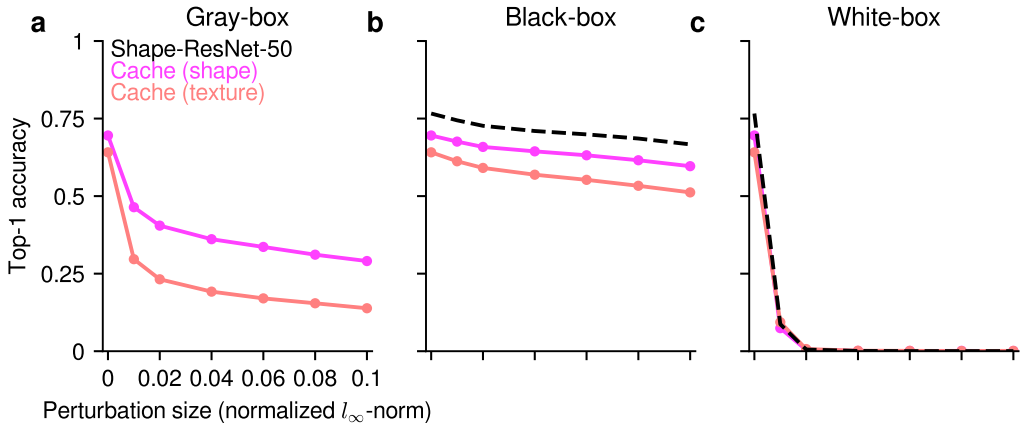

Figure 5: The effect of using Shape-ResNet-50 (shape) vs. ResNet-50 (texture) derived features in the cache on clean and adversarial accuracies in the (**a**) gray-box, (**b**) black-box and (**c**) white-box settings. The results shown here are for the `layer4_bottleneck1_relu` layer.

ResNet-50). It has been argued that Shape-ResNet-50 learns more global, shape-based representations than a standard ImageNet-trained ResNet-50 and it has already been shown to improve robustness on the ImageNet-C benchmark (Geirhos et al., 2019). We confirm this improvement (Table 3; ResNet-50 vs. Shape-ResNet-50) and find that caching with Shape-ResNet-50 leads to roughly the same mCE as the backbone Shape-ResNet-50 itself.

Remarkably, however, when used in conjunction with caching, these Shape-ResNet-50 features also substantially improved the adversarial robustness of the cache models in the gray-box and black-box settings, compared to the ImageNet-trained ResNet-50 features. Figure 5 illustrates this for the `layer4_bottleneck1_relu` cache model. This effect was more prominent for earlier layers.

Shape-based features, however, did not improve the adversarial robustness in the white-box setting, neither for the backbone model nor for the cache models (Table 1). This suggests that eliminating the heavy texture bias of DNNs does not necessarily eliminate the existence of adversarial examples for these models. The opposite, however, seems to be true: that is, adversarially robust models do not display a texture bias; instead they seem to be much more shape-biased, similar to humans (Zhang & Zhu, 2019).

5 Discussion

In this paper, we have shown that a combination of two basic ideas motivated by the cognitive psychology of human vision, an explicit cache-based episodic memory and a shape bias, improves the robustness of image recognition models against both natural and adversarial perturbations at the ImageNet scale. Caching alone improves (gray-box) adversarial robustness only, whereas a shape bias improves natural robustness only. In combination, they improve both, with a synergistic effect in adversarial robustness (Table 4).

Table 4: Summary of our main results. This table is a distilled version of Tables 1 and 3. In each cell, the first number represents the adversarial accuracy (gray-box accuracy for cache models and white-box accuracy for cacheless models, both with $\epsilon = 0.06$); the second number represents the *mCE* score. Note that better models have higher accuracy and lower *mCE* score. Starting from a baseline model with no cache and no shape bias (bottom right), adding a cache memory (bottom left) only improves adversarial accuracy; adding a shape bias (top right) only improves natural robustness; adding both (top left) improves both natural and adversarial robustness with a synergistic improvement in the latter.

	Cache +		Cache -	
Shape bias +	**33.6%**	**73.5**	0.0%	**73.5**
Shape bias -	17.0%	76.4	0.0%	76.4

Why does caching improve adversarial robustness? Orhan (2018) suggested that caching acts as a regularizer. More specifically it was shown in Orhan (2018) that caching significantly reduces the Jacobian norm at test points, which could explain its improved robustness against small-norm perturbations such as adversarial attacks. However, since Jacobian norm only measures local sensitivity, this does not guarantee improved robustness against larger perturbations, such as the natural perturbations in the ImageNet-C benchmark and indeed we have shown that caching, by itself, does not provide any improvement against such perturbations.

It should also be emphasized that caching improves adversarial robustness only under certain threat models. We have provided evidence for improved robustness in the gray-box setting only, Zhao & Cho (2018) and Dubey et al. (2019) also provide evidence for improved robustness in the black-box setting (Orhan (2018) reports evidence for improved robustness through caching in the white-box setting in CIFAR-10 models, however it is likely that such robustness improvements are much easier to achieve in CIFAR-10 models than in ImageNet models). The results in Dubey et al. (2019) are particularly encouraging, since they suggest that the caching approach can scale up in the gray-box and black-box attack scenarios in the sense that larger cache sizes lead to more robust models. On the other hand, neither of these two earlier works, nor our own results point to any substantial improvement in adversarial robustness in the white-box setting at the ImageNet scale. The white-box setting is the most challenging setting for an adversarial defense. Theoretical results suggest that in terms of sample complexity, robustness in the white-box setting may be fundamentally more difficult than achieving high generalization accuracy in the standard sense (Schmidt et al., 2018; Gilmer et al., 2018) and it seems unlikely that it can be feasibly achieved via test-time only interventions such as caching.

Why does a shape bias improve natural robustness? Natural perturbations modeled in ImageNet-C typically corrupt local information, but preserve global information such as shape. Therefore a model that can integrate information more effectively over long distances, for example by computing a global shape representation is expected to be more robust to such natural perturbations. In Shape-ResNet-50 (Geirhos et al., 2019), this was achieved by removing the local cues to class label in the training data. In principle, a similar effect can be achieved through architectural inductive biases as well. For example, Hendrycks & Dietterich (2019) showed that the so-called feature aggregating architectures such as the ResNeXt architecture (Xie et al., 2017) are substantially more robust to natural perturbations than the ResNet architecture, suggesting that they are more effective at integrating local information into global representations. However, it remains to be seen whether such feature aggregating architectures accomplish this by computing a *shape* representation.

In this work, we have also provided important insights into several cache design choices. Scaling up the cache models to datasets substantially larger than ImageNet would require making the cache as compact as possible. Our results suggest that other things being equal, this should be done by clustering the keys rather than by reducing their dimensionality. For very large datasets, the continuous cache retrieval method that uses the entire cache in making predictions (Equations 1 and 2) can be safely replaced with an efficient k-nearest neighbor retrieval algorithm, e.g. Faiss (Johnson et al., 2017), without incurring a large cost in accuracy. Our results also highlight the importance of the backbone choice (for example, Shape-ResNet-50 vs. ResNet-50): in general, starting from

a more robust backbone should make the cache more effective against both natural and adversarial perturbations.

In future work, we are interested in applications of naturally and adversarially robust features in few-shot recognition tasks and in modeling neural and behavioral data from humans and monkeys (Schrimpf et al., 2018).

REFERENCES

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *arXiv preprint arXiv:1802.00420*, 2018.

Timothy F Brady, Talia Konkle, George A Alvarez, and Aude Oliva. Visual long-term memory has a massive storage capacity for object details. *Proceedings of the National Academy of Sciences*, 105 (38):14325–14329, 2008.

Wieland Brendel and Matthias Bethge. Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. *arXiv preprint arXiv:1904.00760*, 2019.

Abhimanyu Dubey, Laurens van der Maaten, Zeki Yalniz, Yixuan Li, and Dhruv Mahajan. Defense against adversarial images using web-scale nearest neigbor search. In *CVPR*, 2019.

Gamaleldin Elsayed, Shreya Shankar, Brian Cheung, Nicolas Papernot, Alexey Kurakin, Ian Goodfellow, and Jascha Sohl-Dickstein. Adversarial examples that fool both computer vision and time-limited humans. In *Advances in Neural Information Processing Systems*, pp. 3910–3920, 2018.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Generalisation in humans and deep neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased toward texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations (ICLR)*, 2019.

Justin Gilmer, Luke Metz, Fartash Faghri, Samuel S Schoenholz, Maithra Raghu, Martin Wattenberg, and Ian Goodfellow. Adversarial spheres. *arXiv preprint arXiv:1801.02774*, 2018.

Edouard Grave, Armand Joulin, and Nicolas Usunier. Improving neural language models with a continuous cache. *arXiv preprint arXiv:1612.04426*, 2016.

Edouard Grave, Moustapha M Cisse, and Armand Joulin. Unbounded cache model for online language modeling with open vocabulary. In *Advances in Neural Information Processing Systems*, pp. 6042–6052, 2017.

Keren Gu, Brandon Yang, Jiquan Ngiam, Quoc Le, and Jonathan Shlens. Using videos to evaluate image model robustness. *arXiv preprint arXiv:1904.10076*, 2019.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations (ICLR)*, 2019.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *arXiv preprint arXiv:1702.08734*, 2017.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Emin Orhan. A simple cache model for image recognition. In *Advances in Neural Information Processing Systems*, pp. 10107–10116, 2018.

Nicolas Papernot and Patrick McDaniel. Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. *arXiv preprint arXiv:1803.04765*, 2018.

Rishi Rajalingham, Elias B Issa, Pouya Bashivan, Kohitij Kar, Kailyn Schmidt, and James J DiCarlo. Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. *Journal of Neuroscience*, 38(33): 7255–7269, 2018.

Jonas Rauber, Wieland Brendel, and Matthias Bethge. Foolbox: A python toolbox to benchmark the robustness of machine learning models. *arXiv preprint arXiv:1707.04131*, 2017. URL `http://arxiv.org/abs/1707.04131`.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115 (3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. In *Advances in Neural Information Processing Systems*, pp. 5014–5026, 2018.

Martin Schrimpf, Jonas Kubilius, Ha Hong, Najib J Majaj, Rishi Rajalingham, Elias B Issa, Kohitij Kar, Pouya Bashivan, Jonathan Prescott-Roy, Kailyn Schmidt, et al. Brain-score: which artificial neural network for object recognition is most brain-like? *BioRxiv*, pp. 407007, 2018.

Lionel Standing. Learning 10000 pictures. *The Quarterly Journal of Experimental Psychology*, 25 (2):207–222, 1973.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. 2013. URL `https://arxiv.org/abs/1312.6199`.

Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1492–1500, 2017.

Tianyuan Zhang and Zhanxing Zhu. Interpreting adversarially trained convolutional neural networks. In *International Conference on Machine Learning*, pp. 7502–7511, 2019.

Jake Zhao and Kyunghyun Cho. Retrieval-augmented convolutional neural networks for improved robustness against adversarial examples. *arXiv preprint arXiv:1802.09502*, 2018.

Zhenglong Zhou and Chaz Firestone. Humans can decipher adversarial images. *Nature communications*, 10(1):1334, 2019.