We appreciate that the reviewers found our work "important" (R2 & R4), "interesting" (R3), "novel" and "relevant to the NeurIPS/CV/AI communities" (R1); found our experiments "well-controlled" (R2) and "quite rigorous and informative" (R4); and found our presentation "clear" (R1, R3, R4). We thank the reviewers for their helpful comments.

Feature difficulty (R3): *"I hope that the authors have a grasp of manually designed image features and their application in the computer vision research area before the deep learning era...Results presented in the paper are all predictable and expected."* We agree that color is an easier feature than shape or texture. However, the goal of our work was **not** to reason about the absolute difficulty of *particular* features (shape, color, texture). Rather, we were testing what deep models come to represent when multiple features are present and correlated to differing degrees with task labels, regardless of what those features are. We performed experiments using both vision and non-vision datasets. Indeed, we found that feature difficulty was not the sole determinant of feature use or representation (Figs. 5 & 6).

Decoding experiments (R1): We tested which features a model represents as a function of training task, which required using a carefully controlled dataset. The joint image feature–label statistics of ImageNet are unknown and uncontrolled. We see our work as complementary to existing work that has looked at feature representations and feature-based classification behavior in ImageNet models (e.g. Geirhos et al. 2019, Brendel and Bethge 2019, Hermann et al. 2019).

Choice of vision model architectures. *Probe a post-AlexNet architecture* (R1): We also performed our vision experiments in ResNet-50, a standard benchmark in computer vision (see Appendix A). *Concern that AlexNet would overfit a small dataset* (R1): It did not in our case. Our validation sets require generalization over held-out features (see Sec. 3). *Choice to reduce FC layer widths* (R1): It is standard practice to reduce classifier sizes in proportion to the number of output classes (see e.g. Qian et al. 2020). Nonetheless, we have done initial experiments with standard FC widths, which show somewhat worse validation performance for texture (and comparable performance for shape and color) compared to the models reported in the paper, when using our current set of hyperparameters.

Connections to the literature. *Theory on learning simple to complex* (R2, R3): The theoretical results of Saxe, Belkin, and others are relevant. However, that work leaves open questions of the effect of multiple redundant features with varying predictivity, which our work attempts to address. We show aspects of feature learning dynamics over training (Supp. Fig. A.9) that would not be completely predicted by Saxe's theory. We will discuss how our work complements prior work in the revision. *Spatial frequency* (R2): We agree that spatial frequency is related to the features we investigate, though we don't see it as equivalent. We will add citations to recent related work (e.g. Yin et al. 2019). *Further discussion of related work on shortcuts, etc.* (R4): We will add further discussion of these papers, which we see as complementary and related. We note that Geirhos et al. investigated classification behavior (not representations), and that dataset statistics influence whether texture bias is observed (see Geirhos et al. 2019, Hermann et al. 2019).

Correlated trifeature datasets (R1, R3): A pair of features (e.g. shape and color) was correlated across the set of images (not within individual images). As an example, suppose that shape and color are correlated with conditional probability = 0.5. Then, in images containing triangles, half would be red and half would be uniformly sampled from the other colors (blue, white, etc.). Similarly, in images containing trapezoids, half would be orange and half would be some other color. The attribute matching (e.g. triangle = red, trapezoid = orange, etc.) was randomly chosen. Although regression decoding is interesting (R3), our experiments also contribute, since classification is a common task.

Binary features datasets (R1): One measure of task difficulty is whether the task is solvable by a single layer, or requires a multi-layer perceptron to solve it, e.g. XOR (Minsky & Papert, 1969). In our (non-vision) Binary Features dataset, we defined two features: one that is learnable by a linear model ("linear" feature, which we call "easy"), and one that requires an MLP (XOR, a "nonlinear" feature, which we call "difficult"). The inputs for this dataset were 32-element binary (1 or 0) vectors in which the first 16 elements instantiated feature A (linear), and the second 16 instantiated feature B (nonlinear). The label was probabilistically determined by these two features — that is, the inputs were sampled so that each feature matched the label a certain percentage of the time. To probe the model's computations, we created new datasets where e.g. the label matched feature B and was uncorrelated with feature A.

Nonlinear decoders (R3): We found similar results on binary features with nonlinear decoders (Supp. Fig. A.10).

Decode from additional model layers (R1 & R2): We considered the output of the convolutional layers because this high-level visual representation is the standard choice for transfer to downstream tasks, and classification layers because these determine the model's classification decision. We agree that it would be interesting to additionally consider earlier model layers in future; we are happy to do so before the camera-ready if the reviewers feel this would be beneficial.

Terminology (R2): We will clarify in the introduction that our "features" are latent variables within the data-generating process underlying the inputs, not necessarily accessible at the pixel level, and that uncovering when and where these latent variables are represented in the model is our goal. We provide some clarification of our definition of difficulty above ("Binary Features Datasets"). We will move our definitions of "suppression" and "enhancement" earlier in text.

RSA (R3): Prior work doesn't perform RSA on models trained on different features. We compare to CKA (Fig. A.13).