Training Vision-Language Transformers from Captions Alone

Liangke Gui[1*] **Qiuyuan Huang**[2]
Alex Hauptmann[1] **Yonatan Bisk**[1,2] **Jianfeng Gao**[2]
[1]Carnegie Mellon University
{liangkeg, alex, ybisk}@cs.cmu.edu
[2]Microsoft Research, Redmond
{qihua, jfgao}@microsoft.com

Abstract

We show that Vision-Language Transformers can be learned without human labels (e.g. class labels, bounding boxes, etc). Existing work, whether explicitly utilizing bounding boxes (1; 2; 3) or patches (4), assumes that the visual backbone must first be trained on ImageNet (5) class prediction before being integrated into a multimodal linguistic pipeline. We show that this is not necessary and introduce a new model **V**ision-**L**anguage from **C**aptions (**VLC**) built on top of Masked Auto-Encoders (6) that does not require this supervision. In fact, in a head-to-head comparison between ViLT, the current state-of-the-art patch-based vision-language transformer which is pretrained with supervised object classification, and our model, **VLC**, we find that our approach 1. outperforms ViLT on standard benchmarks, 2. provides more interpretable and intuitive patch visualizations, and 3. is competitive with many larger models that utilize ROIs trained on annotated bounding-boxes. Code and pretrained models are released at `https://github.com/guilk/VLC`.

1 Introduction

Should vision guide language understanding or does language structure visual representations? Vision-language transformers have put language first. Most popular vision-language transformers (1; 2; 7; 3) only integrate vision from selected bounding boxes extracted by pretrained ImageNet (5) classifiers. In this paradigm, the bag of visual tokens are embedded into an existing linguistic space (i.e. the lexical embeddings of BERT (8)). The introduction of ViT (9) empowered the community to flip the paradigm. Notably, ViLT (4) initializes with ViT (9), so the initial semantic representation is vision based and language must project into the patch space. This flipped paradigm places visual representations as the initial conceptual space to which language must adhere. Additionally, there are engineering benefits to this new paradigm as it removes the computationally expensive need for ROI extraction. However, because ViT is trained with supervised class labels, its representation may be constrained by the limited concepts ImageNet covers and yet the space is still somewhat linguistic in nature when initialized and requires expensive data annotation, a hindrance to scaling to arbitrarily many visual classification categories. We take the important next

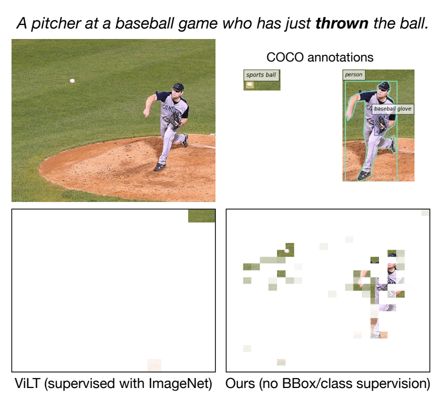


Figure 1: We present an image with its corresponding annotations and caption. Visualized are the model's top aligned patches with the word **thrown**. Note, ViLT often chooses a single (predictive) patch, where our model **VLC** produces a more meaningful (if diffuse) distribution over the relevant patches.

[*] Work done when Liangke interned at Microsoft Research, Redmond.

Preprint. Under review.

step and remove the need for supervised pretraining. A truly unsupervised visual semantics is learned via Masked Auto-Encoders (6) before language is integrated. This leads to both a better performing and more general model. In addition, every component can be improved and scaled with unsupervised and weakly aligned found data – removing the need for future annotation efforts while still scaling to open-vocabulary domains in the wild.

Our **V**ision-**L**anguage from **C**aptions (**VLC**) model matches or outperforms nearly all vision-language transformers despite being 1. Smaller, 2. Avoiding use of ROIs, and 3. Not leveraging supervised pretraining. We evaluate across several popular benchmarks in addition to retrieval and probing. The model performance also appears to continue to improve with data scaling, and as it relies only on weak alignment of image-text pairs, future work with access to large compute may be able to continue driving up performance. Finally, we provide several analyses on the underlying patch/lexical representations to understand what our models are learning and guide future VL transformer research.

2 Related Work

Vision-Language Modeling. Based on how they encode images, most existing works on vision-language modeling fall into three categories. The first category (3; 2; 7; 10; 1; 11; 12; 13; 14) focuses on using pre-trained object detectors to extract region-level visual features (e.g.,by Faster R-CNN (15)). In particular, OSCAR (11) and VinVL (12) further boost the performance by feeding additional image tags into the transformer model. However, extracting region-level features requires pretrained object detectors with high-resolution inputs that can be time-consuming. To tackle these two issues, the second category (16; 17; 18) proposes to encode images by using grid features from convolutional neural networks. SOHO (17) first discretize the grid features by a learnable vision dictionary, and feed the discretized features to their cross-modal module. The third category (19; 20; 21; 22) uses a Vision Transformer (ViT) (9) as the image encoder and design different objective functions for vision-language pretraining. To minimize the computation overhead, ViLT (4) adopts a linear projection layer to encode images, but lags behind the state-of-the-art performance. In our work, we follow ViLT by using a linear projection layer to encode images that is different from previous work with complex ResNe(X)t or object detectors. We investigate how to pretrain a ViT-based model in an end-to-end manner that closes the performance gap while maintaining fast inference speed.

Masked Language Modeling. Masked language modeling (MLM) and its auto-regressive counterparts are widely used in natural language processing for learning text representations. MLM (8) trains a model to predict a random sample of input tokens that have been masked in a multi-class setting. In vision-language modeling, we randomly mask some of the input tokens, and the model is trained to reconstruct the original tokens given the masked tokens and its corresponding visual inputs.

Masked Image Modeling. Masked image modeling (MIM) is a pretext task to learn representations from images corrupted by masking. Inspired by the success of masked language modeling (MLM) in NLP, different masked prediction objectives have been proposed for image tasks. iGPT (23) predicts unknown pixels of a sequence. ViT (9) predicts mean colors of masked patches. BEiT (24) proposes to use a pre-trained discrete variational autoencoder (dVAE) (25) to encode masked patches. MaskFeat (26) predicts HoG (27) features of the masked image regions. SimMIM (28) and MAE (6) predict RGB values of raw pixels by direct regression. MIM has also been explored in the field of vision-language representation learning by either regressing the masked feature values (2; 1; 19; 4) or predicting a distribution over semantic classes for corresponding image region (1; 3; 10).

3 Method

3.1 Model Architecture

Our aim is a parameter-efficient vision-language transformer without the need for supervised pre-training. To this end, we use a ViT-based framework to learn multi-modal representations by 1) intra-modal reconstruction through masked image/language modeling; 2) inter-modal alignment through image-text matching. The architecture of our proposed **VLC** framework is illustrated in Figure 2. **VLC** consists of a modality-specific projection module 3.2, a multi-modal encoder 3.3 and

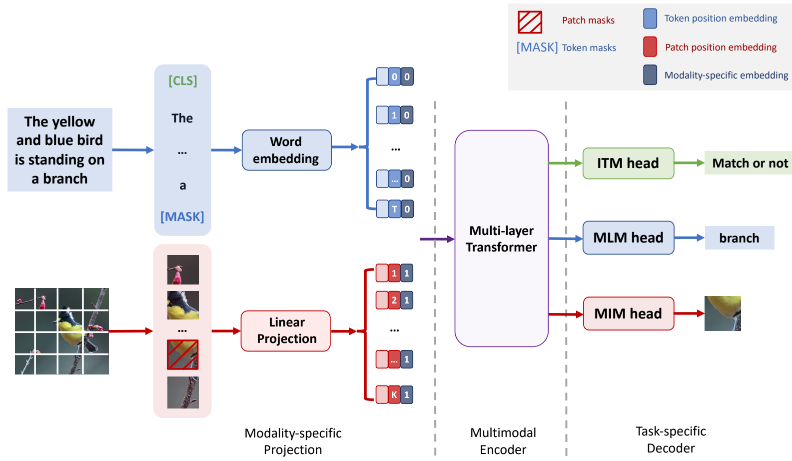


Figure 2: The overall architecture of our **VLC** model. Our model consists of three modules: (1) Modality-specific projection. We use a simple linear projection to embed patched images and a word embedding layer to embed tokenized text; (2) Multi-modal encoder. We use a 12-layer ViT (9) initialized from MAE (6) (ImageNet-1K without labels) as our backbone; (3) Task-specific decoder. We learn our multi-modal representations by masked image/language modeling and image-text matching which are only used during pre-training. We use a 2-layer MLP to fine-tune our multi-modal encoder for downstream tasks. Importantly, we find that the masked image modeling objective is important throughout second-stage pre-training, not only for initialization of the visual transformer.

three task-specific decoders 3.4. We aim for minimal visual and textual embedding designs during pretraining. The red and blue arrows are the information flows of image and text, respectively.

3.2 Modality-specific Projection Module

While most of existing methods rely on complex ResNeXt (16) or object detection components (1; 3; 12; 11), we use a trainable *linear projection* layer to map flattened visual patches to the visual embedding space. The patch embeddings are represented as $\mathbf{v} = \{v_1, ..., v_n\} \in \mathbb{R}^{n \times d}$, where n is the number of image patches and d is the hidden dimension of our model. For text embedder, we follow BERT (29) to tokenize the input sentence into WordPieces (30). We then adopt a word embedding lookup layer to project tokenized words to the textual embedding space. Here we use $\mathbf{w} = \{w_{CLS}, w_1, ..., w_m\} \in \mathbb{R}^{m \times d}$ to represent the token embeddings, where m is the number of tokens and the special token `CLS` denotes the start of the token sequence. We encode patch and token positions separately by $v^{pos} \in \mathbb{R}^{1 \times d}$ and $w^{pos} \in \mathbb{R}^{1 \times d}$. We use $v^{type} \in \mathbb{R}^{1 \times d}$ and $w^{type} \in \mathbb{R}^{1 \times d}$ as modality-type embeddings to distinguish the modality difference between patch and token embeddings. The final representations of each patch v_i and token w_j are calculated as

$$\hat{v}_i = LayerNorm(v_i + v_i^{pos} + v^{type}) \quad \text{and} \quad \hat{w}_j = LayerNorm(w_j + w_j^{pos} + w^{type}). \tag{1}$$

3.3 Multi-modal Encoder

To learn the contextual representations from both visual and textual modality, we follow single-stream approaches (4; 1) and use the ViT-B/16 architecture as our multi-modal encoder. ViT-B/16 consists 12 alternating layers of multiheaded self-attention (MSA) and MLP blocks. LayerNorm comes before every block and residual connections after after every block (9).

We use a merged-attention (19) mechanism to fuse the visual and textual modalities. More specifically, we concatenate the token and patch embeddings together as $\{\hat{w}_{CLS}, \hat{w}_1, ..., \hat{w}_m, \hat{v}_1, ..., \hat{v}_n\}$, then feed them into the transformer blocks to get the contextual representations $\{h_{CLS}, h_1^w, ..., h_m^w, h_1^v, ..., h_n^v\}$. Compared with dual-stream approaches (3; 2; 21; 19), our model design is more parameter-efficient, as the same set of parameters are shared across modalities. As a key difference from existing approaches, we initialize our model with MAE pretrained on ImageNet-1K with no labels.

3.4 Pretraining Objectives

To learn a universal visual and textual representation for vision-and-language tasks, we apply self-supervised methods to pre-train a model on a large aggregated dataset. Unlike previous approaches that only mask text tokens, we randomly mask both image patches and text tokens simultaneously. We train our model with three objectives: masked image modeling (MIM), masked language modeling (MLM) and image-text matching (ITM).

Masked Language Modeling. In language pretraining, MLM randomly masks input tokens, and the model is trained to reconstruct the original tokens based on unmasked context. Following BERT (8), we randomly mask text tokens with a probability of 0.15, and replace the masked ones $\mathbf{w_m}$ with a special token `[MASK]`. The goal is to predict the masked tokens based on both non-masked text tokens $\mathbf{w}_{\backslash\mathbf{m}}$ and image patches $\mathbf{v}_{\backslash\mathbf{m}}$. The learning target $\mathcal{L}_{MLM}$ can be formulated as

$$\mathcal{L}_{MLM} = -\mathbb{E}_{(\mathbf{w},\mathbf{v})\sim D} \log p(\mathbf{w_m}|\mathbf{w}_{\backslash\mathbf{m}}, \mathbf{v}_{\backslash\mathbf{m}}). \tag{2}$$

We use a linear layer with default parameters (29) as the MLM head to output logits over the vocabulary, which are used to compute the negative log likelihood loss for the masked text tokens.

Masked Image Modeling. Existing approaches explore MIM either by regressing the masked features values (1; 4; 20) or by predicting a distribution over semantic classes for a certain image region (1; 3; 19). In contrast, we follow MAE (6) to randomly mask image patches with a probability of 0.6, and reconstruct the missing pixels based on both non-masked tokens $\mathbf{w}_{\backslash\mathbf{m}}$ and patches $\mathbf{v}_{\backslash\mathbf{m}}$. The learning target $\mathcal{L}_{MIM}$ can be formulated as

$$\mathcal{L}_{MIM} = \mathbb{E}_{(\mathbf{w},\mathbf{v})\sim D} f(\mathbf{v_m}|\mathbf{w}_{\backslash\mathbf{m}}, \mathbf{v}_{\backslash\mathbf{m}}), \tag{3}$$

where the feature regression objective f is to regress the masked image patch representations to pixel values. We use 8-layer transformer as the MIM head r. For a masked image patch v_j, the objective f can be formulated as: $f(v_j|\mathbf{w}_{\backslash\mathbf{m}}, \mathbf{v}_{\backslash\mathbf{m}}) = ||r(h_j^v) - v_j||^2$. Each output of the MIM head is a vector of pixel values representing a patch.

Image-Text Matching. Given a batch of image and text pairs, the ITM head identifies if the sampled pair is aligned. We randomly replace the aligned image with a different one with a probability of 0.5. We use the special token `[CLS]` as the fused representation of both modalities, and feed h_{CLS} to the ITM head. The learning target $\mathcal{L}_{ITM}$ can be formulated as

$$\mathcal{L}_{ITM} = -\mathbb{E}_{(\mathbf{w},\mathbf{v})\sim D} \log p(y|\mathbf{w}, \mathbf{v}), \tag{4}$$

Where $y \in \{0, 1\}$ indicates whether the image and text is matched ($y = 1$) or not ($y = 0$). We use a single linear layer as the ITM head and compute negative log likelihood loss as our ITM loss.

We weight the pretraining objectives equally so the full pre-training objective is:

$$\mathcal{L} = \mathcal{L}_{MLM} + \mathcal{L}_{ITM} + \mathcal{L}_{MIM} \tag{5}$$

4 Experiments and Results

We conduct extensive experiments on a diversified set of vision-language benchmarks, including image-text retrieval, visual question answering and natural language for visual reasoning. We evaluate our pretrained model to each downstream task through end-to-end fine-tuning. To further show the generalization ability of our pre-trained model, we examine our model on ImageNet-1K classification task following common practice (9; 6). For a fair comparison, we compare our model with state-of-the-art methods on the base model size.

4.1 Pre-training Datasets

Following previous work (1; 4; 21; 19), our pre-training corpus comprises four commonly used vision-language datasets including COCO (31), Visual Genome (32), Google Conceptual Captions (33) and SBU Captions (34), totalling 4.0M unique images and 5.1M image-text pairs. To show the benefits of data-scaling, we also compare to the VinVL (12) pretraining setting which includes Flickr30k (35), GQA (36), VQA (37), VG-QAs (32) and a subset of OpenImages (38). This larger pre-training corpus contains 5.65M unique images (see detailed statistics in Appendix A.1). Future work can trivially grow the size of the corpus by including large-scale web crawls.

Model	Params	Text Retrieval						Image Retrieval					
		Flickr30K (1K)			MSCOCO (5K)			Flickr30K (1K)			MSCOCO (5K)		
		@1	@5	@10	@1	@5	@10	@1	@5	@10	@1	@5	@10
ALBEF (21)	187M	94.3	99.4	99.8	73.1	91.4	96.0	82.8	96.7	98.4	56.8	81.5	89.2
VinVL (12)	157M	-	-	-	74.6	92.6	96.3	-	-	-	58.1	83.2	90.1
UNITER (1)	155M	85.9	97.1	98.8	64.4	87.4	93.1	72.5	92.4	96.1	50.3	78.5	87.2
OSCAR (11)	155M	-	-	-	70.0	91.1	95.5	-	-	-	54.0	80.8	88.5
PixelBERT (16)	144M	87.0	98.9	99.5	63.6	87.5	93.6	71.5	92.1	95.8	50.1	77.6	86.2
ViLT (4)	87M	83.5	96.7	98.6	61.5	86.3	92.7	64.4	88.7	93.8	42.7	72.9	83.1
VLC (ours – 5.6M)	87M	**89.2**	**99.2**	**99.8**	**71.3**	**91.2**	**95.8**	**72.4**	**93.4**	**96.5**	**50.7**	**78.9**	**88.0**

Table 1: We compare **VLC** to ViLT on text-image retrieval, as they have are patch based and have the same number of parameters. We see substantial gains across all settings. For a complete comparison, we include several state of the art bounding box based and supervised methods. ALBEF, the largest, outperforms all models, but our approach is nonetheless competitive in most settings.

4.2 Downstream Tasks

Visual Question Answering (VQA (37)). Given an input image and a question, the VQA task is to predict an answer from the visual content. We conduct experiments on VQAv2 dataset (37) that is built on MSCOCO. It contains 83K images for training, 41K for validation, and 81K for testing. We report performance on the test-dev and test-std splits. Following previous work (2; 1; 21), we use the training, validation splits and additional question-answer pairs from Visual Genome while reserving $1,000$ validation image-question pairs for internal validation.

Natural Language for Visual Reasoning (NLVR^2 (39)). Given a triplet of two images and a description, this task is to predict whether this description describes a pair of images. Following previous work (4; 1), we use the *pair* method which treats one input sample as two image-text pairs by repeating the text twice. Each pair is passed through our model and we take the concatenation of two pooled representation `[CLS]` from our model as the representation of one input sample.

Image-Text Retrieval. Image-Text retrieval contains two subtasks: image-to-text retrieval (TR) and text-to-image retrieval (IR). We evaluate our pre-trained models on the Karpathy splits (40) of MSCOCO (31) and Flickr30K (41) in fine-tuning settings. MSCOCO contains 123K images, and each image has five corresponding human-written captions. We split the data into 82K/5K/5K training/validation/test images. To be consistent with previous work (4; 1), we use the additional 30K images from MSCOCO validation set to improve the performance. Flickr30K contains 31K images with five captions for each image. We split the data into 30K/1K/1K as the training/validation/test set.

4.3 Implementation Details

The multi-modal encoder uses a 85.8M parameter ViT-B/16 architecture initialized with MAE pre-trained on ImageNet-1K without labels. For text inputs, we tokenize text with the *bert-base-uncased* tokenizer. The text embedding parameters are learned from scratch, in lieu of loading pre-trained BERT weights. We randomly mask image patches with a probability of 0.6 and text tokens with a probability 0.15. To accelerate training, we follow MAE (6) and skip the mask token `[MASK]` in the encoder and only apply it in the lightweight decoder. We use AdamW (42) with a weight decay of 0.01. The learning rate is warmed-up to $1e^{-4}$ in the first 10% of total training steps and is decayed to zero for the rest of the training following a linear schedule. During pre-training, we resize the shorter edge of input images to 384, take random image crops of resolution 384×384, and apply RandAugment (43) with the hyper-parameters of $N = 2, M = 9$. We pre-train for $200k$ steps with a batch size of $4,096$ on 128 NVIDIA V100 GPUs that takes 80 hours. For all downstream tasks, we fine-tune for 10 epochs with a batch size of 256 for VQAv2/retrieval tasks and 128 for NLVR^2. For the parameter estimation, we exclude the textual embedder as it is shared by all vision-language transformers following ViLT (4). We also exclude the parameters of all the auxiliary heads as they are only required during pretraining. More implementation details can found in Appendix A.2.

4.4 Adapt VLC to Domain-Specific Tasks and Evaluation

Image-Text Retrieval Tasks. We begin with a proof of concept experiment, evaluating our model on the Karpathy splits of the Flickr30K (41) and MSCOCO (31) benchmarks. In Table 1, we compare

Model	Params	VQAv2 test-dev	VQAv2 test-std	NLVR^2 dev	NLVR^2 test
Supervised ImageNet Bounded Boxes					
ViLBERT (3)	274M	70.55	70.92	-	-
LXMERT (2)	240M	72.42	72.54	74.90	74.50
VisualBERT (7)	170M	70.80	71.00	67.4	67.0
UNITER (1)	155M	72.70	72.91	77.18	77.85
OSCAR (11)	155M	73.16	73.44	78.07	78.36
VinVL (12)	157M	**75.95**	**76.12**	**82.05**	**83.08**
Supervised ImageNet Classes					
ALBEF* (21)	187M	**74.54**	**74.70**	**80.24**	**80.50**
Visual Parsing (20)	180M	74.00	74.17	77.61	78.05
PixelBERT (16)	144M	74.45	74.55	76.5	77.2
ViLT (4)	87M	71.26	-	75.70	76.13
No supervised classes or bounding boxes					
VLC (ours – 4M)	87M	72.98	73.03	77.04	78.51
VLC (ours – 5.6M)	87M	**74.02**	**74.0**	**77.70**	**79.04**

Table 2: Comparison with our model with state-of-the-art pre-trained methods on vision-language understanding tasks. Our model (**VLC**), unlike all others, is only pre-trained with weakly-aligned image-caption pairs. Again, our approach outperforms ViLT (the closest comparison model) and is competitive with larger and more heavily supervised approaches. Rows are highlighted in shades of gray to mark use of bounding boxes and ImageNet classes. *ALBEF uses an additional 6-layer 81M parameter transformer decoder to generate answers on the VQA task which increases the model parameters to around 270M.

several strong multimodal transformers in the literature which leverage ROIs, more parameters, and are pretrained on ImageNet classification. Note that as most of detection-based models have the advantage of using Faster R-CNN (15) pre-trained on VG (32) or MSCOCO (31). ALBEF uses a pre-trained ViT-B/16 and BERT model as their backbone which doubles the model size. Additionally, they specifically design the coarse-to-fine objectives while we directly fine-tune the pre-trained ITM head for retrieval tasks. Thus we treat ALBEF as a strongest available baseline.

The closest comparison to our approach is ViLT as it is the same model size, though still requires more supervised data in the form of ImageNet classification pretraining for ViT (9)[2]. In addition, UNITER uses a frozen object detector and a trainable BERT model as their backbone which has a comparable model size. We can see substantial gains on both tasks compared with UNIERT.

Image-Text Understanding Tasks. Table 2 presents **VLC** results on two popular image-text understanding datasets: VQAv2 and NLVR^2. For VQAv2, we report the test-dev and test-std scores returned from the evaluation server. For NLVR^2, we evaluate our models on both *dev* and *test-P* split.

Comparison to models supervised/initialized with ImageNet bounded boxes. Most of these models use object detectors pretrained on VG (32) or MSCOCO (31) to extract region features. Object detectors help in VQA tasks as they mainly ask about objects. Within the similar scale of pretraining data, our model achieves competitive performance on both tasks. Note that our model uses 384×384 or 576×576 as input resolution during our fine-tuning stages. This resolution is much lower compared with previsou work using 800×1333 (3; 1). In particular, *VinVL* (12) has a multi-stage pre-training for its object detector that has access to ImageNet-5K (44) (6.8M images from 5K classes) and four object detection datasets (45; 32; 31; 38) (2.5M images with bounding box annotations). The most comparable appraoch is UNITER as we use the same training data (*i.e.*, 4M images) and trainable parameters. Our approach performs better than UNITER which uses a pretrained object detector and BERT as initialization.

Comparison to models with supervised ImageNet classes. Most of these approaches use additional visual embedders together with a pretrained BERT as their backbones. For example, ALBEF (21), Visual Parsing (20), PixelBERT (16) use pre-trained ViT-B/16, Swin transformer, ResNeXt-152 as

[2]ViLT uses ViT-B/32 pretrained with ImageNet-21K and finetuned on ImageNet-1K with supervised labels.

their visual embedder, respectively. All these embedders are trained with supervised ImageNet-1K. In addition, ALBEF uses a 6-layer transformer decoder to generate answers on the VQA task which further increases the model size. With the same pretraining data, our approach outperforms ViLT by 1.72% on the VQA test-dev split, 1.34% and 2.38% on NLVR^2 dev and test split. We further verify the scalability of our model by using the same scale pre-training data as VinVL. Experiments show that our model achieves comparable results with larger and more heavily supervised approaches.

4.5 Ablation Study

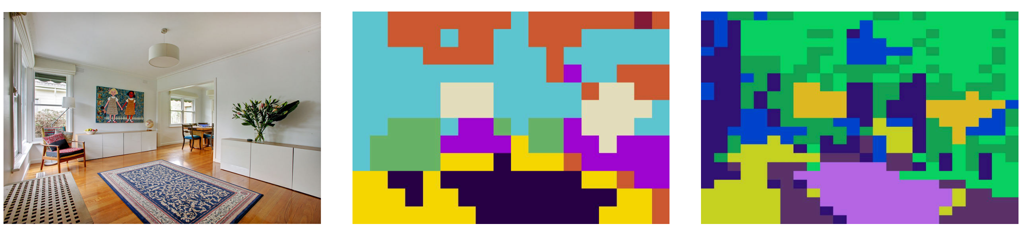
To understand the impact of different components, we ablate and compare variants of our model (*i.e.*, pretraining objectives and image resolutions used during fine-tuning) and report VQAv2 test-dev accuracy in Table 3. Note that UNITER without pre-training feeds object detections to a pretrained BERT (accuracy is copied from their Table 2 Line 2). Comparing models without vision-language pretraining, theirs outperforms ours by 3.94% which indicates pretrained object detectors and BERT are strong priors. Our experiments show that both MIM and ITM are important throughout pretraining, which contrasts to findings in previous work (19; 4; 21).

	Train Objective			**Resolution**			**VQAv2**
	MIM	MLM	ITM	384^2	480^2	576^2	**Acc.**
UNITER*							69.39
Base **VLC**				√			65.45
		√	√	√			69.06
Objective	√	√		√			68.98
	√	√	√	√			69.52
Resolution	√	√	√		√		69.69
	√	√	√			√	**69.97**

Table 3: Ablation study on objectives and image resolutions (top lines show model performance without pretraining). Our experiments show that both image and text masked modeling improve the performance. Additionally, there is a consistent improvement by increasing the image resolution during fine-tuning. *The input image size of UNITER is 800×1333.

5 Understanding the models

While simpler and more efficient, patch-based models differ in important ways from traditional bounding-box based approaches. In particular, while the visual stack is traditionally frozen in those models, now the entire “backbone" is learnable. Also, where previously, the goal was to “map" vision to language, now the two are learned jointly. We therefore take this opportunity to investigate the models to better understand how their behaviors differ due to the two (pre-)training objectives.



Original Image ViLT clusters **VLC** clusters

Figure 3: Visualization of patch clusters for an example image as produced from ViLT (many densely clustered patches) versus **VLC**’s more fine-grained and diffuse representations.

Understanding Patches. We begin with a simple patch clustering visualization (Figure 3). Without the inclusion of any language, we can simply cluster (and color) the visual patch embeddings of ViLT and **VLC**. ViLT relies on on larger patches (32×32) for higher resolution (384×640). We instead use smaller patches and lower resolution (16! $\times$16 for 384×384). It is also easy to see how both models are identifying key semantic regions of the image (e.g. the rug, painting and plant). Also note, both models incorrectly place the painting and plant in the same cluster.

To investigate this representation collapse at scale, we leverage the nocaps dataset (46). Nocaps provides captions for images based on object classes in COCO, similar to COCO, and out of domain. By visualizing the embedding similarities of nouns from these three classes with patches in the images, we can determine: 1. Are ViLT patches more tightly clustered – perhaps due to the discriminative training objective and 2. How do both models’ behaviors change for classes more (or less) like the ImageNet pretraining. In Figure 4, we see several trends. First, ViLT’s “most similar" patch to

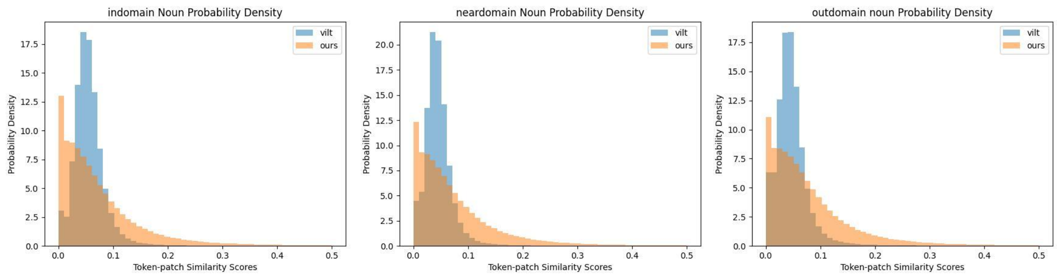


Figure 4: These plots are the top noun-patch similarity per image as produced by both ViLT and **VLC**. ViLT rarely produces a high similarity lexical score, likely due to its discriminative pretraining objective and its score distribution shifts down as we move further away from its supervised pretraining data. In contrast, **VLC** has a much smoother distribution and high lexical alignment across all settings.

the noun rarely has a passes 0.1, perhaps indicating that they are not shifting from their pretrained representations. Second, we see the mass shift slightly lower as we move from left to right (in-domain to out-of-domain), indicating the model has a harder time finding alignments to novel words. **VLC** has a markedly different behavior, with a smoother overall set of similarities – often able to to find a visual patch with high similarity to the query across all conditions. **VLC** also exhibits an opposite trend where the model's scores climb as we shift out of domain. These plots do *not* show if the alignment is semantically meaningful, but they do show starkly different behaviors. This concentration of embeddings by ViLT can also be seen visually in examples in the Appendix A.3.

Image Classification. Given that the underlying visual representations are shifting through the cross-modal training, we run a simple image classification experiment to see the effects language training has on the underlying visual "backbone". We compare **VLC** with state-of-the-art models on ImageNet-1K classification and report top-1 validation accuracy of a single 384×384 crop. During end-to-end fine-tuning on ImageNet-1K, we follow the supervised ViT training procedure: AdamW (42) with a base learning rate of $1e^{-3}$ and a weight decay of 0.05, batch size of 1024, and the first 5 epochs are used as warm up and decayed to $1e^{-6}$ following a cosine schedule. As our model is not pre-trained with a discriminative loss, we use a global pooling of encoder outputs as image representations.

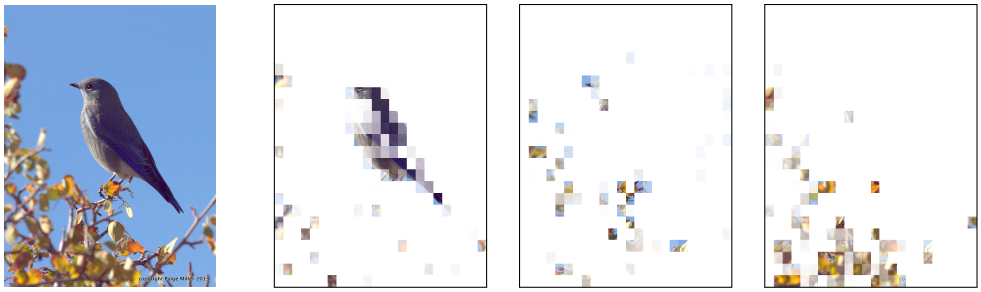
As shown in Table 4, **VLC** learns generic representations which are transferable to vision tasks. With only fine-tuning on ImageNet-1K, our model matches the performance of Swin-B (48) that is trained with supervised labels. Note that BEiT (24) is a two-stage pre-training model of which the tokenizer is trained on 250M examples of DALLE (25) data. Compared with MAE (6), our model learns competitive multi-modal representations from vision-language pre-training while retains high-quality image representations.

Model	Size	Top-1
Supervised		
ViT-B/16 (9)	384^2	77.9
DeiT-B (47)	384^2	83.1
Swin-B (48)	384^2	**84.5**
Self-supervised		
DINO (49)	224^2	82.8
MoCo v3 (50)	224^2	83.2
MaskFeat (26)	224^2	83.6
SimMIM (28)	224^2	83.8
BEiT* (24)	384^2	**84.6**
MAE (6)	224^2	83.6
VLC	384^2	**84.5**

Table 4: Models are pretrained on ImageNet 1K and self-supervised models are evaluated by end-to-end fine-tuning. *BEiT uses a DALLE (25) pre-trained tokenizer.

6 Visualizations

These patch-language transformer architectures allow for intuitive visualizations of the lexical alignment. Doing so provides a simple way to explore what the model is learning to represent about an image. In Figure 5 we show results from visualizing three different words in the same caption for an image from COCO. Not that for the word branch, the model is actively attempting to avoid the abundant leaves. Second, since there is nothing about our model besides the MAE initialization that should be biased (as shown previously) towards ImageNet classes, we present three images in Figure 6 that highlight words not present in the standard ImageNet1K training split used by other models. Specifically, a noun (*string*), adjective (*yellow*), and verb (*swinging*). These demonstrate the general trend of ViLT often focusing on surprising locations.



The bird is on the branch with leaves alone | bird | branch | leaves

Figure 5: Lexical-Patch alignment for an image in MS COCO. We visualize three different words from the same caption to see how the model uniquely represents them. This is a particularly challenging case as the model attempts to isolate patches for branches separate from those with leaves.

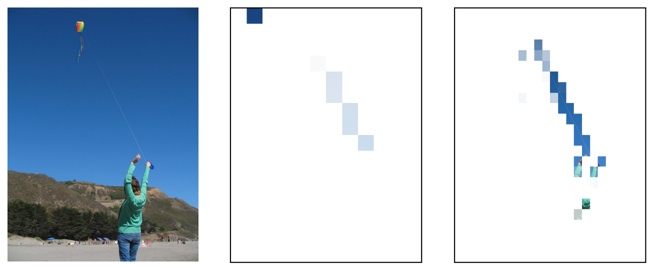
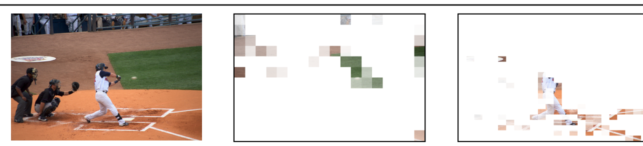
Caption with focus	Original Image / ViLT / **VLC**
A person on a beach holding a kite string and a kite is in the air	
A cat sitting on a chair, that is blue and yellow	
A baseball player swinging a baseball bat at a baseball	

Figure 6: To investigate concepts not present in COCO or ImageNet, we present three images and highlighted words which are out of domain (i.e. not in ImageNet-1K). Specifically, we are visualizing a noun (top), adjective (middle) and verb (bottom). The model again delicately avoids nearby but distinct concepts (e.g. the cat on the chair or irrelevant parts of the baseball field). More examples and analysis can be found in Appendix A.3.

7 Conclusion

We present a VLP architecture, **V**ision-**L**anguage from **C**aptions (**VLC**), pretrained with image-caption pairs. While **VLC** only uses a linear projection layer as the image embedder, it achieves competitive performance on a diversified set of vision-language tasks to existing approaches that rely on object detectors or supervised CNN/ViT networks. We also evaluated the effectiveness of our vision-language pretraining on ImageNet-1K classification task to show that **VLC** retains high-quality image representations. Finally, our visualization demonstrates that **VLC** can accurately align image patches with text tokens and the performance scales with increased training data. This opens an exciting door to large scale weakly supervised open-domain vision-and-language models.

References

[1] Chen, Y.-C., L. Li, L. Yu, et al. Uniter: Universal image-text representation learning. In *ECCV*. 2020.

[2] Tan, H., M. Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *EMNLP*, 2019.

[3] Lu, J., D. Batra, D. Parikh, et al. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *NeurIPS*, 2019.

[4] Kim, W., B. Son, I. Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *ICML*. 2021.

[5] Russakovsky, O., J. Deng, H. Su, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015.

[6] He, K., X. Chen, S. Xie, et al. Masked autoencoders are scalable vision learners. *CVPR*, 2022.

[7] Li, L. H., M. Yatskar, D. Yin, et al. Visualbert: A simple and performant baseline for vision and language. *ACL*, 2019.

[8] Devlin, J., M.-W. Chang, K. Lee, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*. 2019.

[9] Dosovitskiy, A., L. Beyer, A. Kolesnikov, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021.

[10] Su, W., X. Zhu, Y. Cao, et al. Vl-bert: Pre-training of generic visual-linguistic representations. *ICLR*, 2020.

[11] Li, X., X. Yin, C. Li, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. *ECCV*, 2020.

[12] Zhang, P., X. Li, X. Hu, et al. Vinvl: Making visual representations matter in vision-language models. *CVPR*, 2021.

[13] Li, W., C. Gao, G. Niu, et al. Unimo: Towards unified-modal understanding and generation via cross-modal contrastive learning. *ACL*, 2021.

[14] Qi, D., L. Su, J. Song, et al. Imagebert: Cross-modal pre-training with large-scale weak-supervised image-text data. *arXiv preprint arXiv:2001.07966*, 2020.

[15] Ren, S., K. He, R. Girshick, et al. Faster r-cnn: Towards real-time object detection with region proposal networks. *NeurIPS*, 2015.

[16] Huang, Z., Z. Zeng, B. Liu, et al. Pixel-bert: Aligning image pixels with text by deep multi-modal transformers. *arXiv preprint arXiv:2004.00849*, 2020.

[17] Huang, Z., Z. Zeng, Y. Huang, et al. Seeing out of the box: End-to-end pre-training for vision-language representation learning. In *CVPR*. 2021.

[18] Jia, C., Y. Yang, Y. Xia, et al. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*. 2021.

[19] Dou, Z.-Y., Y. Xu, Z. Gan, et al. An empirical study of training end-to-end vision-and-language transformers. *CVPR*, 2022.

[20] Xue, H., Y. Huang, B. Liu, et al. Probing inter-modality: Visual parsing with self-attention for vision-and-language pre-training. *NeurIPS*, 2021.

[21] Li, J., R. Selvaraju, A. Gotmare, et al. Align before fuse: Vision and language representation learning with momentum distillation. *NeurIPS*, 2021.

[22] Radford, A., J. W. Kim, C. Hallacy, et al. Learning transferable visual models from natural language supervision. In *ICML*. 2021.

[23] Chen, M., A. Radford, R. Child, et al. Generative pretraining from pixels. In *ICML*. 2020.

[24] Bao, H., L. Dong, F. Wei. Beit: Bert pre-training of image transformers. *ICLR*, 2022.

[25] Ramesh, A., M. Pavlov, G. Goh, et al. Zero-shot text-to-image generation. In *ICML*. 2021.

[26] Wei, C., H. Fan, S. Xie, et al. Masked feature prediction for self-supervised visual pre-training. *arXiv preprint arXiv:2112.09133*, 2021.

[27] Dalal, N., B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*. 2005.

[28] Xie, Z., Z. Zhang, Y. Cao, et al. Simmim: A simple framework for masked image modeling. In *CVPR*. 2022.

[29] Devlin, J., M.-W. Chang, K. Lee, et al. Bert: Pre-training of deep bidirectional transformers for language understanding. *NAACL*, 2019.

[30] Wu, Y., M. Schuster, Z. Chen, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

[31] Lin, T.-Y., M. Maire, S. Belongie, et al. Microsoft coco: Common objects in context. In *ECCV*. 2014.

[32] Krishna, R., Y. Zhu, O. Groth, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 123(1):32–73, 2017.

[33] Sharma, P., N. Ding, S. Goodman, et al. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*. 2018.

[34] Ordonez, V., G. Kulkarni, T. Berg. Im2text: Describing images using 1 million captioned photographs. *NeurIPS*, 2011.

[35] Young, P., A. Lai, M. Hodosh, et al. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL*, 2:67–78, 2014.

[36] Hudson, D. A., C. D. Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*. 2019.

[37] Goyal, Y., T. Khot, D. Summers-Stay, et al. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*. 2017.

[38] Krasin, I., T. Duerig, N. Alldrin, et al. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://github.com/openimages*, 2016.

[39] Suhr, A., S. Zhou, A. Zhang, et al. A corpus for reasoning about natural language grounded in photographs. *ACL*, 2018.

[40] Karpathy, A., L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*. 2015.

[41] Plummer, B. A., L. Wang, C. M. Cervantes, et al. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*. 2015.

[42] Loshchilov, I., F. Hutter. Decoupled weight decay regularization. In *ICLR*. 2018.

[43] Cubuk, E. D., B. Zoph, J. Shlens, et al. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPR Workshop*. 2020.

[44] Xie, S., R. Girshick, P. Dollár, et al. Aggregated residual transformations for deep neural networks. In *CVPR*. 2017.

[45] Shao, S., Z. Li, T. Zhang, et al. Objects365: A large-scale, high-quality dataset for object detection. In *ICCV*. 2019.

[46] Agrawal, H., K. Desai, Y. Wang, et al. nocaps: novel object captioning at scale. In *ICCV*. 2019.

[47] Touvron, H., M. Cord, M. Douze, et al. Training data-efficient image transformers & distillation through attention. In *ICML*. 2021.

[48] Liu, Z., Y. Lin, Y. Cao, et al. Swin transformer: Hierarchical vision transformer using shifted windows. In *CVPR*. 2021.

[49] Caron, M., H. Touvron, I. Misra, et al. Emerging properties in self-supervised vision transformers. In *CVPR*. 2021.

[50] Chen, X., S. Xie, K. He. An empirical study of training self-supervised vision transformers. In *ICCV*. 2021.

[51] Clark, K., M.-T. Luong, Q. V. Le, et al. Electra: Pre-training text encoders as discriminators rather than generators. In *ICLR*. 2020.

A Appendix

This supplementary material has three sections. Section A.1 describes the details of our pretraining datasets. Section A.2 describes our implementation details for downstream tasks. Section A.3 shows more visualization examples with more comparisons.

A.1 Pre-training dataset

The statistics of the pretraining dataset is shown in Table 5. Most of the existing approaches, such as UNITER (1) and ViLT (4), use MSCOCO, VG, GCC and SBU to pre-train their models. We denote this training set as *base*. To verify the scalability of our model, we follow VinVL (12) to further incorporate VQA, VG-QA, GQA, Flickr30K and OpenImages. As there are some overlaps among VG, MSCOCO and VQA, we exclude all those training images that appear in the downstream tasks via URL matching.

Dataset	MSCOCO	VG	GCC	SBU	VGA	GQA	VG-QA	Flickr30K	OpenImages
# Images	113K	100K	2.95M	860K	83K	79K	87K	29K	1.67M
# Text	567K	769K	2.95M	860K	545K	1026K	931K	145K	1.67M

Table 5: Statistics of the pre-training dataset

A.2 Implementation Details for Downstream Tasks

For all downstream tasks, we fine-tune our model with a learning rate of $5e^{-4}$ for 10 epochs. We use a layer-wise learning rate decay (51) of 0.5. We use 576×576 as the input image resolution for the VQA task and 384×384 for NLVR^2 and image-text retrieval tasks.

Visual Question Answering. We use a 2-layer MLP with a hidden size of $1,536$ to adapt **VLC** to the VQA task. We follow the standard practice (4) to convert the task to a multilabel classification task with $3,192$ answer classes. Following previous work (1; 21), we use additional question-answer pairs from VG for data augmentation. We select additional question-answer pairs if the corresponding images and answers appear in the VQA train and validation splits.

Natural Language for Visual Reasoning. . As there are two input images and a single description, we follow OSCAR (11), ViLT (4) and VinVL (12) by using the *pair* method. Similar to the settings of the VQA task, we use a 2-layer MLP with a hidden size of $1,536$ to adapt **VLC** to the NLVR^2 task.

Image-Text Retrieval. We conduct experiments on both MSCOCO and Flickr30K datasets. Given an image, we use the corresponding text as a positive example while randomly sample 15 text as negative examples. We use a fully connected layer as our retrieval similarity head that is initialized from the pre-trained ITM head. We fine-tune our model with a cross-entropy loss to maximize the probabilities on positive pairs.

A.3 Analysis on More Examples

We show additional examples for nouns in Figure 7, adjectives in Figure 8, and verbs in Figure 9.

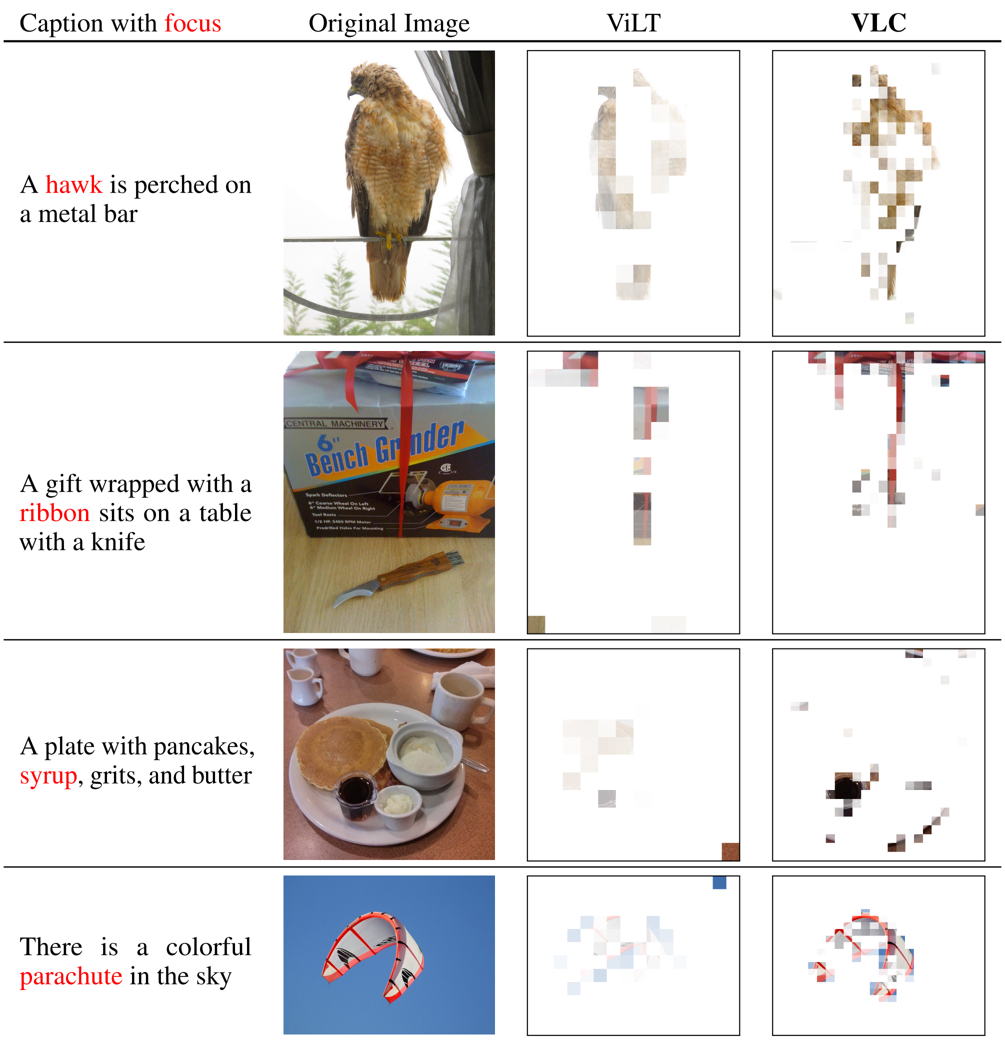


Figure 7: Visualized are OOD noun examples. Note that ViLT is often picking up on relevant features but has a single strongest correlation with a single, presumably predictive, patch.

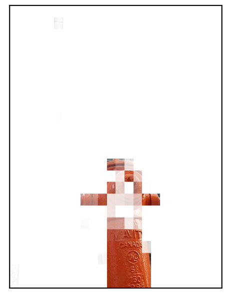
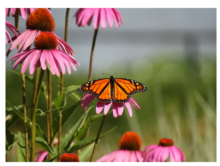
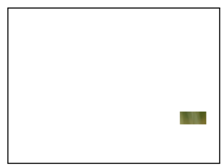
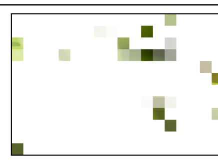
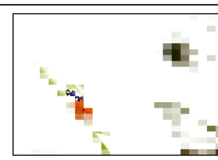
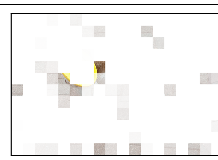
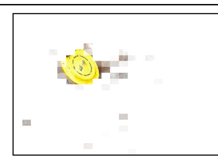
Caption with focus	Original Image	ViLT	**VLC**
A red fire hydrant in front of a skyscraper			
A monarch butterfly lands on a pink flower.			
A small orange and blue ladybug sitting on long green leaves			
A brown and white dog is holding a yellow Frisbee			

Figure 8: Visualized are OOD adjective examples. **VLC** produces more accurate and comprehensive masks. Note that the lady bug is correctly identified but not exclusively and likely not based on an understanding of the relative size *small*. Future work would ideally show results that indicate models understanding more abstract and comparative concepts.

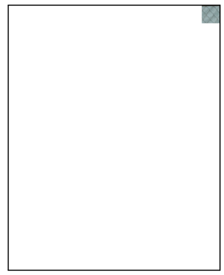
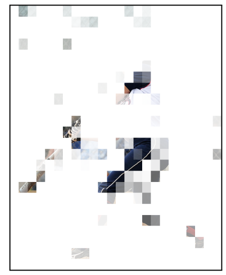
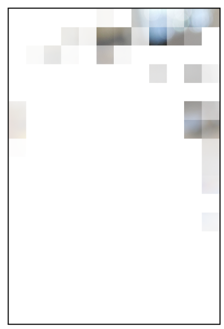
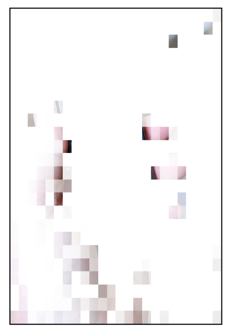
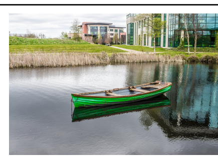
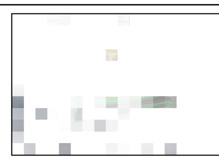
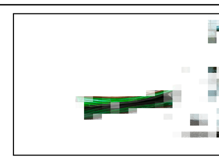
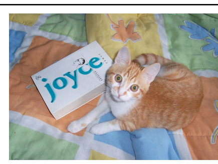
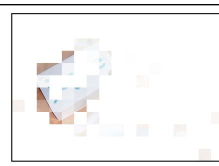
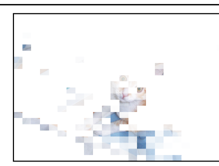
Caption with focus	Original Image	ViLT	**VLC**
A person who is hitting a ball with a bat.			
A person holding a cell phone in their hand			
A green boat floating on top of a body of water			
an orange and white cat sitting on a bed staring at the viewer			

Figure 9: Visualized are OOD verb examples. Note that verbs from still images is a slightly strange concept, but there are key perceptual indicators that align to the verb's semantics. For example, *holding* is aligned to the person's hands and *staring* picks up on the cat's eyes.