CLIPSym: Delving into Symmetry Detection with CLIP

Tinghan Yang
Purdue University
yang1683@purdue.edu

Md Ashiqur Rahman
Purdue University
rahman79@purdue.edu

Raymond A. Yeh
Purdue University
rayyeh@purdue.edu

Abstract

Symmetry is one of the most fundamental geometric cues in computer vision, and detecting it has been an ongoing challenge. With the recent advances in vision-language models, i.e., CLIP, we investigate whether a pre-trained CLIP model can aid symmetry detection by leveraging the additional symmetry cues found in the natural image descriptions. We propose CLIPSym, which leverages CLIP's image and language encoders and a rotation-equivariant decoder based on a hybrid of Transformer and G-Convolution to detect rotation and reflection symmetries. To fully utilize CLIP's language encoder, we have developed a novel prompting technique called Semantic-Aware Prompt Grouping (SAPG), which aggregates a diverse set of frequent object-based prompts to better integrate the semantic cues for symmetry detection. Empirically, we show that CLIPSym outperforms the current state-of-the-art on three standard symmetry detection datasets (DENDI, SDRW, and LDRS). Finally, we conduct detailed ablations verifying the benefits of CLIP's pre-training, the proposed equivariant decoder, and the SAPG technique.

1. Introduction

Symmetry plays an important role in human perception and understanding of the world [6, 8, 13, 30, 45, 58]. In computer vision, symmetry is one of the most fundamental geometric cues, providing essential information for tasks such as object recognition [35, 57], scene understanding [10, 64], image matching [16], and editing [33]. Detecting symmetry, however, has been a long-standing question in computer vision due to the variations and complexities present in real-world scenarios [19, 31, 42, 49, 56, 66].

Earlier works relied on keypoint matching techniques [1, 3, 36, 52, 53], which involved comparing local descriptors of keypoints and their transformed counterparts. Although effective to some extent, these methods struggled with complex symmetry patterns or in the presence of noise. More recently, deep learning-based approaches [12, 50, 51] were proposed to detect reflection and rotation symmetries and have shown promising results. PMCNet [50] proposed a method that relies on specially designed convolutional techniques rather than principled equivariant architectures, limiting its ability to consistently detect symmetry patterns across different orientations. Although EquiSym [51] addresses this limitation by leveraging group-equivariant convolutional networks, due to the limited availability of large-scale annotated symmetry datasets, the full potential of the learning based approach remains underexplored.

On the other hand, recent advances in pre-trained vision-language foundation models [20, 37, 40, 73], have shown remarkable generalization capabilities by leveraging large-scale datasets and joint training on visual and textual information. Image captions often contain words or phrases that carry the symmetry information of the object in the image. For example, in the case of internet-scale LAION-400M dataset [48], around 10% of the image captions contain words that convey shape/symmetry-related cues such as 'rectangle,' 'circle,' 'oval', *etc*. (see Appendix A2.1 for detailed statistics). This observation suggests that vision-language models trained on such extensive image-text pairs are likely to contain useful symmetry cues in their learned text/image representations. A natural question arises:

> *How to leverage pre-trained vision-language models for symmetry detection?*

In this paper, we present CLIPSym, a novel framework that leverages the pre-trained CLIP model for the detection of reflection and rotation symmetries in images. Our approach is motivated by the hypothesis that being trained on a large-scale vision-language dataset, the visual representations learned by CLIP contain knowledge that can benefit symmetry detection. The proposed CLIPSym contains CLIP's image and text encoders and introduces a decoder with guarantees on rotation equivariance. Given an input image, CLIPSym outputs a symmetry heatmap, where each pixel represents the probability of being a reflection axis or the rotation center. To fully leverage the potential of CLIP's language encoder, we propose a novel Semantic-Aware Prompt Grouping (SAPG) method, which aggregates multiple text prompts to enhance the model's understanding of symmetries.

To validate our method's performance, we conduct a set of comprehensive experiments on three symmetry detection datasets and observe that our proposed CLIPSym achieves state-of-the-art performance for both reflection and rotation symmetry detection tasks. Finally, we analyze our model's equivariance properties and conduct ablations for each of the proposed components.

Our contributions are as follows:

- We introduce CLIPSym, a framework that, for the first time, leverages the multimodal understanding abilities of CLIP to achieve end-to-end detection of reflection and rotation symmetries.
- We propose SAPG, a novel prompting technique to enhance the model's understanding of symmetries through the aggregation of a diverse set of prompts.
- We propose a symmetry decoder with theoretical guarantees for rotation equivariance, which improves the model's robustness to diverse symmetry patterns.
- We demonstrate that CLIPSym achieves state-of-the-art performance across multiple benchmark datasets. Extensive ablation studies further validate the importance of CLIP pre-training, the SAPG technique, and the equivariant decoder.

2. Related work

Symmetry detection. Earlier works tackled the task of symmetry detection primarily through *keypoint matching*, which works by comparing local features of corresponding key points between an image and its mirrored version. Often, techniques such as spatial and angular auto-correlation are employed [22, 24, 26]. Local feature descriptors, such as SIFT [51], couture, and edge features [1, 36, 52, 59, 60], are frequently utilized to achieve a degree of equivariance to image transformations and detection of boundaries of symmetric objects.

Symmetry detection can also be formulated as a dense prediction task by assigning a score to each pixel of the image. Tsogkas and Kokkinos [55] employed a bag of features and multiple-instance learning in their model. On the other hand, Gnutti et al. [14] computed a symmetry score for each pixel using patch-wise correlation and gradient for validating candidate axes. Fukushima and Kikuchi [11], Funk and Liu [12] used data-driven learning-based approaches for symmetry detection. Polar matching convolution (PMC) [50] is used to attain higher reflection consistency in symmetry detection. To achieve perfect rotation and translation equivariance, Seo et al. [51] used group equivariant CNNs to predict per-pixel symmetry scores.

While existing methods have achieved promising performance in symmetry detection, they still face challenges in modeling diverse symmetry patterns and lack large annotated datasets. In this paper, we aim to overcome these limitations by leveraging the power of the pre-trained CLIP model, which has learned visual-semantic representations with generalization capability to real-world scenes.

Equivariant networks. Equivariance to geometric transformations in input images constitutes a vital inductive bias, fostering improved generalization and consistency, particularly under conditions of limited training data. While Convolutional Neural Networks (CNNs) inherently exhibit equivariance to the translation operations, achieving equivariance to a broader spectrum of geometric transformations is not guaranteed. This broader family of equivariance is achievable through Group Equivariant CNNs [4] and parameter sharing strategies [21, 70, 72]. Notably, Steerable CNNs [5, 61–63] offer an efficient approach by representing filters in terms of steerable bases. Recent works have extended the scope of equivariance to include diverse transformations, such as scaling [38, 54, 65], sampling [39, 46], color changes [25], permutation [15, 28, 29, 43, 70, 71, 75], and extending beyond CNN architectures to encompass Vision Transformers [47, 68]. Equivariance is particularly important in our task of symmetry detection, as it allows the model to consistently identify symmetrical patterns regardless of their orientation or position in the image, leading to more robust and accurate predictions.

Vision & language models. CLIP [37], a seminal pre-trained vision-language model, has gathered significant attention and has been widely adopted in various downstream tasks, including monocular depth estimation [18], sound source localization [34], scene text detection and spotting [69], video understanding [41], semantic segmentation [32], etc. Recently, prompting has emerged as a prominent paradigm for efficiently adapting pre-trained models to downstream tasks. Zhou et al. [78] and Zhou et al. [77] propose methods to automatically learn prompt tokens that yield strong performance on target tasks. Khattak et al. [23] introduce a multi-modal prompting approach to effectively adapt CLIP to various applications. Furthermore, Bahng et al. [2] explores the use of visual prompts to probe CLIP's visual representation learning capabilities.

3. Approach

We propose CLIPSym, a model that leverages the pre-trained CLIP model (ViT-B/16) for the task of symmetry detection. Given an input of an image $\boldsymbol{I} \in \mathbb{R}^{H \times W \times 3}$, CLIPSym outputs the predicted symmetry heatmap $\hat{\boldsymbol{S}}_{\boldsymbol{I}} \in [0, 1]^{H \times W}$, which represents the probability of each pixel being reflection axes or the rotation center for objects in $\boldsymbol{I}$. As CLIP has been trained on an internet-scale dataset, we hypothesize that such pre-training would be beneficial to symmetry detection. The main challenge is how to build a model that utilizes this pre-trained knowledge.

To leverage the image information from CLIP, we use the pre-trained image encoder $E_{\texttt{img}}$ to extract image features

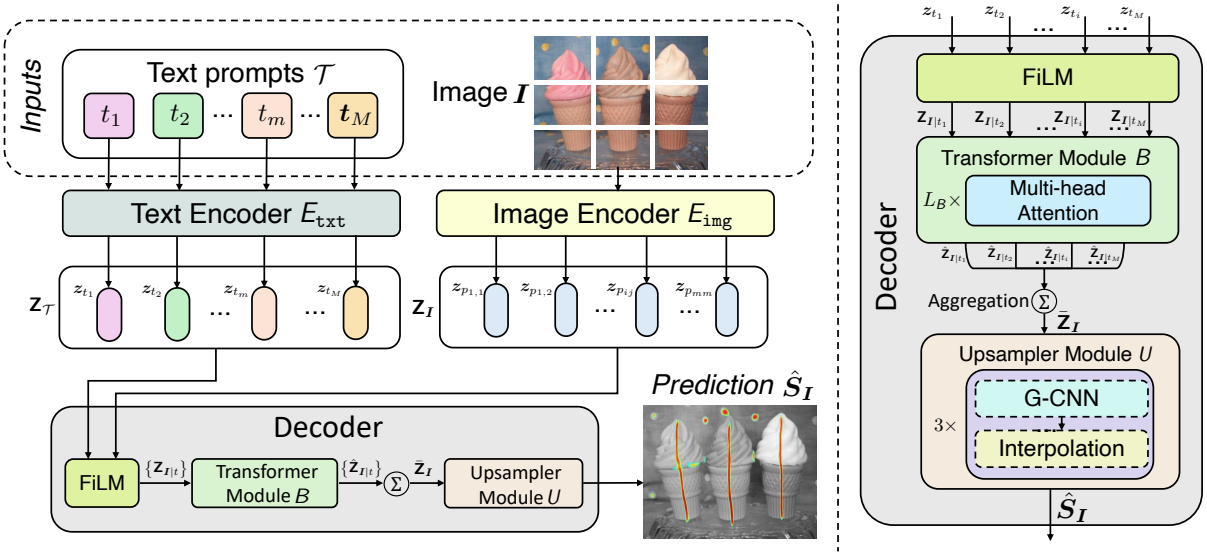


Figure 1. Overview of the proposed CLIPSym architecture. **Left:** The text encoder $E_{\texttt{txt}}$ encodes prompts in set $\mathcal{T}$ as $\mathbf{Z}_{\mathcal{T}}$ and the image encoder $E_{\texttt{img}}$ encodes patches in image I as $\mathbf{Z}_I$. $\mathbf{Z}_I$ and $\mathbf{Z}_{\mathcal{T}}$ are then mixed and aggregated in the decoder to get the final predicted symmetry heatmap $\hat{S}_I$. **Right**: Visualization of decoder details.

from the set of image patches tokens $\mathbf{Z}_I = \{\boldsymbol{z}_{p_{ij}}\}$, where $\boldsymbol{z}_{p_{ij}} \in \mathbb{R}^d$ denotes the image feature of patch p_{ij} at position $(i,j) \in \mathbb{Z}_M^2$, with M representing the number of patches along each dimension. To leverage the text information from CLIP, we design SAPG, which integrates a set of text prompts $\mathcal{T} = \{t_1, t_2, \dots\}$ and use the text encoder $E_{\texttt{txt}}$ to extract a set of text tokens $\mathbf{Z}_{\mathcal{T}} = \{\boldsymbol{z}_{t_1}, \boldsymbol{z}_{t_2}, \dots\}$ where each $\boldsymbol{z}_{t_i} \in \mathbb{R}^d$.

With the image and text tokens extracted, we then propose a *rotation equivariant decoder* to mix and aggregate the tokens into a final heatmap. A visual overview of our approach is illustrated in Fig. 1. We will now discuss the decoder details in Sec. 3.1, followed by the prompting technique SAPG in Sec. 3.2, and training details in Sec. 3.3.

3.1. Rotation equivariant decoder

Decoder architecture. The decoder module D takes the set of the image tokens $\mathbf{Z}_I$ and text tokens $\mathbf{Z}_{\mathcal{T}}$ as inputs and generates the final symmetry heatmap *i.e.*, $\hat{S}_I$; an overview is provided in Fig. 1 (left). Our proposed decoder module consists of three modules, namely, a FiLM block, a Transformer module followed by aggregation, and finally, a rotation equivariant upsampler. We design the decoder to be rotation equivariant, as prior work [51] has shown equivariance guarantees to benefit the performance of symmetric detection. We now discuss each of the building blocks.

① *FiLM block:* A FiLM [9] conditioning layer utilizes the text tokens to modulate the image features, allowing image tokens to carry textual semantic information. For each text token $\boldsymbol{z}_t \in \mathbf{Z}_{\mathcal{T}}$, the FiLM layer generates a set of image tokens modulated by text condition t:

$$\mathbf{Z}_{I|t} = \texttt{FiLM}(\boldsymbol{z}_t, \mathbf{Z}_I) \quad (1)$$

$$= \{\boldsymbol{z}_{p_{ij}|t} | (i,j) \in \mathbb{Z}_M^2 \ , \ \boldsymbol{z}_{p_{ij}|t} \in \mathbb{R}^d\}, \quad (2)$$

where each $\boldsymbol{z}_{p_{ij}|t}$ is computed as

$$\boldsymbol{z}_{p_{ij}|t} = \gamma(\boldsymbol{z}_t) \odot \boldsymbol{z}_{p_{ij}} + \beta(\boldsymbol{z}_t). \quad (3)$$

Here, $\odot$ denotes element-wise multiplication between text and image patch features, and $\gamma(\cdot), \beta(\cdot)$ are linear layers.

② *Transformer module & aggregation:* With the set of image tokens modulated for each text t, we then use a Transformer module B to further learn the spatial dependencies between patches, which is crucial for detecting global symmetry structures. The Transformer module consists of several multi-headed attention blocks, each containing a self-attention layer and multi-layer perceptron (MLP) layers followed by layer normalization as described in ViT [7].

Each set of text-modulated image tokens $\mathbf{Z}_{I|t}$ is passed to the transformer module B to obtain the set of updated tokens

$$\hat{\mathbf{Z}}_{I|t} = \{\hat{\boldsymbol{z}}_{p_{ij}|t} | (i,j) \in \mathbb{Z}_M^2\} = B(\mathbf{Z}_{I|t}) \ \forall t \in \mathcal{T}. \quad (4)$$

Next, we aggregate across all text prompts to construct the set of final tokens $\bar{\mathbf{Z}}_I$ via a weighted average:

$$\bar{\mathbf{Z}}_I = \{\bar{\boldsymbol{z}}_{p_{ij}} | (i,j) \in \mathbb{Z}_M^2\}$$

$$\text{where each } \bar{\boldsymbol{z}}_{p_{ij}} = \sum_{t \in \mathcal{T}} \boldsymbol{w}_t \hat{\boldsymbol{z}}_{p_{ij}|t}. \quad (5)$$

Here, $\boldsymbol{w}_t \in \mathbb{R}$ is a weight scalar corresponding to prompt t therefore $\boldsymbol{w} \in \mathbb{R}^{|\mathcal{T}|}$ learns to combine the patch-conditioned

tokens. The weights satisfy $\boldsymbol{w}_t \geq 0$ and $\sum_{t=1}^{|\mathcal{T}|} \boldsymbol{w}_t = 1$. The upsampler will next process these tokens.

③ *Rotation equivariant upsampler:* To achieve equivariance, we choose to use steerable G-Conv [5] and choose G to be roto-translation group $\mathbb{Z}_M^2 \rtimes C_n$, where C_n denotes a group of $360°/n$ rotations and $n = 4k$, where $k \in \mathbb{Z}^+$.

As G-Conv takes a feature map on a group as input, we first need to convert the set of aggregated tokens $\bar{\mathbf{Z}}_{\boldsymbol{I}}$ to a grid and then lift it to the roto-translation group. Recall, each element $\bar{z}_{p_{ij}} \in \bar{\mathbf{Z}}_{\boldsymbol{I}}$ has a corresponding spatial location p_{ij}. We put back (`Grid`) the elements into a 2D feature map as:

$$\mathbf{F} \triangleq \texttt{Grid}(\bar{\mathbf{Z}}_{\boldsymbol{I}}) \in \mathbb{R}^{d \times M \times M}$$
$$\text{where } \mathbf{F}[:, i, j] = \bar{z}_{p_{ij}} \ \forall\, (i,j) \in \mathbb{Z}_M^2. \tag{6}$$

We then lift this feature map $\mathbf{F}$ to the roto-translation group. The lifted feature map $\mathbf{F}^{\uparrow} \in \mathbb{R}^{|C_n| \times d' \times m \times m}$ is defined as

$$\mathbf{F}^{\uparrow} \triangleq \texttt{Concat}\left([\boldsymbol{R}_\theta \mathbf{F};\ \forall \theta \in C_n]\right), \tag{7}$$

where $\boldsymbol{R}_\theta$ denotes rotation on $2D$ plane. In more details,

$$\mathbf{F}^{\uparrow}[i, \theta, x, y] = \mathbf{F}[i, x', y'] \tag{8}$$
$$\text{where } \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}. \tag{9}$$

More compactly, we denote this action as $[x', y'] = \mathbf{r}_{-\theta}(x, y)$.

The lifted feature map $\mathbf{F}^{\uparrow}$ is then passed through 3 layers of G-Conv [4] and $4\times$ bi-linear upsampling,where G-conv computes the following:

$$(\mathbf{F}^{\uparrow} \star_G \psi)[\theta, x, y] = \sum_{\theta' \in G} \sum_{(x', y') \in \mathbb{Z}_M^2}$$
$$\mathbf{F}^{\uparrow}[\theta', x', y'] \psi[\theta' - \theta, \mathbf{r}_{-\theta}[(x' - x, y' - y)]. \tag{10}$$

Finally, the feature on the roto-translation group is mean-pooled along the rotation dimension θ in the last layer to generate the final prediction symmetry heatmap $\hat{\boldsymbol{S}}_{\boldsymbol{I}}$.

Rotation equivariance guarantees. We will now show that our proposed decoder D is rotation equivariant to the group C_4, *i.e.*, a 2D "rotation" on the image tokens $\mathbf{Z}_{\boldsymbol{I}}$ leads to the same rotation of the prediction $\hat{\boldsymbol{S}}_{\boldsymbol{I}}$. We define a "rotation" using the action $\boldsymbol{T}_\theta$ on the set of patch-features $\mathbf{Z}_{\boldsymbol{I}}$ as

$$\boldsymbol{T}_\theta \mathbf{Z}_{\boldsymbol{I}} \triangleq \{\boldsymbol{T}_\theta \boldsymbol{z}_{p_{ij}}\} = \{\boldsymbol{z}_{p_{\pi_\theta(ij)}}\}, \tag{11}$$

where π_θ rotates the $2D$ coordinates, *i.e.*, a permutation on the patch location (i, j).

Claim 1. *The decoder D is rotation $(\boldsymbol{T}_\theta, \boldsymbol{R}_\theta)$-equivariant to C_4,* i.e.,

$$\mathsf{D}(\boldsymbol{T}_\theta \mathbf{Z}_{\boldsymbol{I}}, \mathbf{Z}_{\mathcal{T}}) = \boldsymbol{R}_\theta \hat{\boldsymbol{S}}_{\boldsymbol{I}} \ \forall \theta \in C_4. \tag{12}$$

Proof. It is sufficient to prove that each component of the decoder is equivariant.

① *FiLM block:* The FiLM block performs element-wise affine transformation (multiplication and addition) for each of the patch features separately. As any operation performed individually on each element of the set is permutation equivariant [75], i.e., $\texttt{FiLM}(\boldsymbol{T}_\theta \mathbf{Z}_{\boldsymbol{I}}, \boldsymbol{z}_t) = \boldsymbol{T}_\theta \mathbf{Z}_{\boldsymbol{I}|t}$.

② *Transformer module & aggregation:* In C_4, $\theta \in \{n \cdot 90° : n \in \mathbb{Z}\}$ then π_θ acts as a permutation on the patch location (i, j) and the action $\boldsymbol{T}_\theta$ can be described as a permutation on the patch features.

$$\texttt{Grid}(\boldsymbol{T}_\theta \mathbf{Z}_{\boldsymbol{I}}) = \boldsymbol{R}_\theta[\texttt{Grid}(\mathbf{Z}_{\boldsymbol{I}})]. \tag{13}$$

Transformer layers are equivariant to permutation on the order of the tokens [74]. So, the application of transformer block B is equivariant, *i.e.* $\mathsf{B}(\boldsymbol{T}_\theta \mathbf{Z}_{\boldsymbol{I}|t}) = \boldsymbol{T}_\theta \mathsf{B}(\mathbf{Z}_{\boldsymbol{I}|t})$. The aggregated of tokens $\bar{\mathbf{Z}}_{\boldsymbol{I}}$ is constructed by a weighted average over the text prompts. As the tokens are spatially aligned, the aggregated token remains equivariant.

③ *Rotation equivariant upsampler:* The upsampler U and relative interpolations are equivariant to C_n, where n is a multiple of 4 by design. As C_4 is a subgroup of C_n, the upsampler is equivariant to C_4, *i.e.*,

$$\mathsf{U}(\texttt{reshape}(\boldsymbol{T}_\theta \bar{\mathbf{Z}}_{\boldsymbol{I}})) = \mathsf{U}(\boldsymbol{R}_\theta \mathbf{F}) = \boldsymbol{R}_\theta \mathsf{U}(\mathbf{F}) = \boldsymbol{R}_\theta \hat{\boldsymbol{S}}_{\boldsymbol{I}}. \tag{14}$$

This concludes the proof. □

3.2. Semantic-Aware Prompt Grouping (SAPG)

While commonly used text prompts such as "`a photo of a [CLASS]`" seems to be a good choice, symmetry is a highly abstract concept that almost exists across a variety of objects, making it unlikely for CLIP's pre-training data to include specific descriptions like "symmetry axes" or "rotation centers". In CLIPSym, we propose a novel prompting technique SAPG to address this challenge. SAPG constructs a set of prompts $\mathcal{T} = \{t_1, t_2, \cdots, t_M\}$, with each prompt t_m representing a string of the combination of K frequent object classes that appear in the dataset, which are separated by spaces. Formally:

$$t_m = \text{"}[\texttt{obj}_{m_1}]\ [\texttt{obj}_{m_2}]\ \cdots\ [\texttt{obj}_{m_K}]\text{"}, \tag{15}$$

where $\texttt{obj}_{m_k}$ represents the k-th object class in the m-th prompt. For example, with $K = 3$ objects in each prompt, the prompt t_m can be "`apple cloud table`". Note that we use the same prompt set $\mathcal{T}$ for all images, ensuring a consistent semantic initialization across the dataset. More details of prompts are provided in Appendix A1.

The design of SAPG is motivated by three key insights:

- *Better initialization via frequent objects:* Since pre-trained CLIP has good language-image alignment, using frequent objects as prompts leads the model to focus more on regions where symmetry is naturally present, thus allowing

the model to learn the underlying symmetric structures rather than dealing with noisy or inconsistent semantic signals from less common words.

- *Aggregation for prompts:* Grouping multiple prompts allows the model to leverage complementary semantic cues such that the model can capture broader aspects of symmetry than a single prompt, which typically focuses on only limited aspects of symmetry. Moreover, the aggregated embeddings of grouped prompts are refined during training, which provides a more robust representation of symmetry.
- *Fixed prompts for a universal concept:* As the concept of symmetry is universal, this means its core characteristics do not vary significantly from one image to another, it is reasonable to use a fixed set of prompts rather than adapting them for each image. In this way, the model has a more consistent semantic anchor that reflects symmetry. Moreover, although prompts are fixed, their embeddings are continuously updated during training, allowing them to gradually evolve to capture the essential characteristics of symmetry more accurately.

In our experiments, we explore various strategies for selecting object classes. Detailed prompt design and more discussions on the motivations of language are in Appendix A1 and A2.

3.3. Model training

In symmetry detection, the class imbalance problem arises due to the low ratio of foreground pixels indicating the rotation/reflection axis to the background pixel. To address this issue, we follow prior works [27, 50] to utilize the $\alpha-$focal loss defined as

$$\mathcal{L}_{\text{focal}}(\boldsymbol{I}) = \sum_{x,y} -\alpha'_{\boldsymbol{I}_{xy}}(1-\hat{\boldsymbol{S}}'_{\boldsymbol{I}_{xy}})^{\lambda}\log(\hat{\boldsymbol{S}}'_{\boldsymbol{I}_{xy}}), \tag{16}$$

where $\hat{\boldsymbol{S}}'_{\boldsymbol{I}_{xy}}$ represents the predicted heatmap for image $\boldsymbol{I}$ at position (x, y), $\alpha'_{\boldsymbol{I}_{xy}}$ represents the symmetry/non-symmetry class balance factor calculated from a pre-defined scalar α, and λ denotes the focusing parameter. Detailed definitions can be found in the Appendix A4.

We fine-tune from the pre-trained CLIP text and image encoders. This decision is driven by two key considerations. First, CLIP has been pre-trained on a vast corpus of image-text pairs, but its training objective does not specifically focus on symmetry detection. Second, the prompts are aimed at capturing the abstract concept of symmetry rather than specific object classes. Hence, it requires fine-tuning the text encoder to map these prompts to text tokens for symmetry detection.

4. Experiments

For a fair comparison, we strictly follow the evaluation protocol of prior works [50, 51]. We first discuss the experimental setup, followed by the results, and conclude with a set of ablation studies.

4.1. Experimental setup

Dataset. As in prior works [51], we conduct experiments on the task of reflection and rotation symmetry detection using three datasets: DENDI [51], SDRW [50], and LDRS [50].

The DENDI dataset consists of 2493 and 2079 images annotated for reflection axes and rotation centers, with 1750/374/369 and 1459/313/307 images in train/validation/test splits for reflection and rotation symmetry, respectively. On the other hand, SDRW and LDRS are reflection datasets that have 51/-/70 and 1110/127/240 images in train/-validation/test splits. Although the original SDRW dataset includes both rotation and reflection data, we only use its reflection data because its rotation data has already been incorporated into DENDI.

Baselines. We compare our approach with three baseline methods considered by Seo et al. [51], including SymResNet [12], PMCNet [50], and EquiSym [51]. SymResNet applied ResNet [17] to detect reflection and rotation symmetries using human-labeled annotated data. PMCNet proposed polar matching convolution to detect reflection symmetries by leveraging polar feature pooling and self-similarity encoding. EquiSym introduced a group-equivariant convolutional network to detect reflection and rotation by utilizing equivariant feature maps, surpassing the performance of all previous methods.

Beyond existing works, we further consider additional baselines: CLIPSym$^{\text{no-text}}$ only uses a CLIP image encoder followed by the same *equivariant decoder* as CLIPSym without any text conditioning, which will reflect the benefits of language. CLIPSym$^{\text{scratch}}$ trains the proposed model from scratch instead of using the pre-trained CLIP, which aims to show that pre-training is helpful. CLIPSym$^{\text{non-eq.}}$ is a variant of CLIPSym with a non-equivariant decoder that uses standard CNN blocks for upsampling, which will show the importance of the design of the equivariant decoder.

Evaluation metrics. To evaluate the symmetric detection tasks, we report the F1-score following Seo et al. [51]. We also report robustness and consistency metrics with respect to rotation and reflection transformations for each of the models. The evaluation metrics are summarized below:

F1-score (↑) is formally calculated as

$$\text{F1} = \max_{\tau}\left(\frac{2\cdot\text{precision}_{\tau}\cdot\text{recall}_{\tau}}{\text{precision}_{\tau}+\text{recall}_{\tau}}\right), \tag{17}$$

where $\tau \in [0, 1]$ is used to threshold the predicted score map at various levels in the range of $[0, 1]$ to obtain binary maps. For each threshold, we compute the F1-score by comparing the predictions against the ground-truth binary heatmaps at the pixel level. The max F1-score across all thresholds is reported as the final performance measure.

Method	Pre-training	Reflection F1	Rotation F1
SymResNet* [12]	ImageNet	30.7	11.9
PMCNet* [50]	ImageNet	52.0	–
PMCNet [50]	ImageNet	53.8 ± 0.5	–
EquiSym* [51]	ImageNet	64.5	22.5
EquiSym [51]	ImageNet	61.7 ± 0.6	22.0 ± 0.7
CLIPSym$^{\text{no-text}}$	ImageNet	54.8 ± 0.2	9.0 ± 0.1
CLIPSym$^{\text{no-text}}$	CLIP	63.7 ± 0.3	17.7 ± 0.2
CLIPSym$^{\text{scratch}}$	–	32.1 ± 0.2	4.7 ± 0.2
CLIPSym$^{\text{non-eq.}}$	CLIP	62.9 ± 0.2	24.2 ± 0.1
CLIPSym$^{\text{eq.}}$	CLIP	$\mathbf{66.5 \pm 0.2}$	$\mathbf{25.1 \pm 0.1}$

Table 1. Quantitative comparison of F1-score (%) on the DENDI dataset [51]. Results of SymResNet*, PMCNet*, and EquiSym* are obtained from the EquiSym [51] paper, while PMCNet and EquiSym are reproduced using the publicly available code. As SymResNet [12] does not have publicly available code, we are unable to report its standard deviation.

Robustness-score assesses the model's robustness under transformations, including reflections and rotations, which is calculated as the F1-score on the transformed dataset. During the assessment of rotation robustness, we sample rotation angles uniformly distributed between $[-45°, 45°]$ and apply them to images in the dataset and their relative ground-truth heatmaps. For reflection robustness, we randomly apply a horizontal flip on each image.

Consistency-score is defined as the cross-entropy loss between the transformed model's outputs and the model's output on the transformed input images. Formally,

$$\text{Consistency} = \frac{1}{|\mathcal{D}|} \sum_{I \in \mathcal{D}} \mathbb{E}_{\boldsymbol{T}} \Big[\text{CE}\big(\boldsymbol{T}(\hat{\boldsymbol{S}}_{\boldsymbol{I}}), \hat{\boldsymbol{S}}_{\boldsymbol{T}(\boldsymbol{I})}\big) \Big], \quad (18)$$

where $\boldsymbol{T}$, defined in Eq. (11), denotes the transformation (*e.g.*, rotation or reflection), CE denotes the cross-entropy function between the two symmetry heatmaps. A lower score indicates a higher consistency, suggesting the model can maintain more consistent predictions faced with reflection or rotation transformations.

Implementation details. As the backbone network, we adopt the pre-trained CLIP model [37] with a ViT-B/16 structure. The model is trained for 500 epochs using the Adam optimizer. To meet the input requirements of the image encoder, training images are reshaped to 417×417 resolution by resizing original images while maintaining the aspect ratio and padding if necessary. During testing, images are reshaped using the same process, where the predictions are cropped and resized to the original image sizes before computing metrics. See Appendix A4 for more details.

4.2. Results

Quantitative results. In Tab. 1 and Tab. 2, we present the F1-score of baseline models pre-trained on different datasets or trained from scratch for detecting reflection and rotation symmetries on DENDI and reflection symmetry on SDRW and LDRS datasets, respectively.

Method	Pre-training	SDRW F1	LDRS F1	Mixed F1
PMCNet [50]	ImageNet	40.8 ± 0.4	30.5 ± 0.5	33.8 ± 0.2
EquiSym [51]	ImageNet	48.2 ± 0.1	37.7 ± 0.1	41.1 ± 0.1
CLIPSym$^{\text{no-text}}$	ImageNet	31.3 ± 0.1	25.3 ± 0.1	27.0 ± 0.1
CLIPSym$^{\text{no-text}}$	CLIP	46.8 ± 0.2	36.2 ± 0.1	39.7 ± 0.1
CLIPSym$^{\text{scratch}}$	–	10.8 ± 0.3	10.4 ± 0.2	10.8 ± 0.3
CLIPSym$^{\text{non-eq.}}$	CLIP	47.8 ± 0.3	37.0 ± 0.1	40.8 ± 0.2
CLIPSym$^{\text{eq.}}$	CLIP	$\mathbf{51.8 \pm 0.3}$	$\mathbf{39.5 \pm 0.1}$	$\mathbf{42.8 \pm 0.1}$

Table 2. F1-score of reflection symmetry detection on SDRW, LDRS, and their mixed datasets.

Method	Pre-training	Robustness ↑	Consistency ↓
PMCNet [50]	ImageNet	52.2	0.417
EquiSym [51]	ImageNet	57.1	0.244
CLIPSym$^{\text{non-eq.}}$	CLIP	58.3	0.093
CLIPSym$^{\text{eq.}}$	CLIP	**59.7**	**0.082**

Table 3. Equivariance robustness and consistency evaluation results for DENDI reflection dataset under $[-45°, 45°]$ uniformly distributed rotation operations.

From Tab. 1 and Tab. 2, we observe the following:

CLIPSym achieves SOTA performance. In Tab. 1, we observe that CLIPSym has the highest F1 score across both tasks, outperforming EquiSym* by 2.0% and 2.6%, the previous SOTA, on the DENDI dataset.

CLIP's pre-training is helpful. In both Tab. 1 and Tab. 2, we observe that CLIPSym pre-trained on CLIP significantly outperforms CLIPSym trained from scratch. CLIPSym without text conditioning pretrained on CLIP also outperforms those pretrained on ImageNet, which suggests that pre-training on a larger and more diverse dataset is beneficial.

CLIPSym effectively leverages the information from the text encoder. This can be seen from the comparison between CLIPSym$^{\text{no-text}}$ and CLIPSym, where CLIPSym outperforms its counterpart in all settings. This suggests that the text encoder provides additional contextual information that helps the model to understand symmetries better.

Beyond performance, we further study how equivariance plays a role in the models. In Tab. 3, we present the Consistency and Robustness score of models on the DENDI reflection dataset. Here, we report the consistency and robustness of rotations uniformly randomly sampled within ± 45 degrees. Interestingly, we observe CLIPSym$^{\text{non-eq.}}$ with a non-equivariant decoder surpasses the two compared baselines in consistency and robustness. Note that both EquiSym [51] and our CLIPSym use the C_8 group-equivariant convolutions, which are only equivariant at intervals of $45°$. This again highlights the importance of CLIP's pre-training in the encoder for consistent image representations. Finally, the full CLIPSym with an equivariant decoder further im-

Method	PMCNet [50]	EquiSym [51]	CLIPSym (ours)
GFLOPs	167.7	**114.0**	148.8

Table 4. Comparions of computation cost in GFLOPs.

proves the consistency and robustness of the model. Please refer to Appendix A3.3 for consistency and robustness results on the SDRW and LDRS datasets.

Comparisons on the computational cost. We report the computational costs in GFLOPs as in Tab. 4 for each of the baselines and our method. We observe that CLIPSym has a slightly higher computational cost at 148.8 GFLOPs compared to EquiSym (114.0 GFLOPs), but is more efficient than PMCNet (167.7 FLOPs). That is, CLIPSym achieves a significant performance improvement over other baselines at a moderate increase in computation.

Qualitative results. In Fig. 2a and Fig. 2b, we compare the predicted reflection and rotation heatmaps of different models and the ground truth. We observe that CLIPSym generates sharper and more accurate symmetry heatmaps compared to the baselines.

In Fig. 3, we present heatmaps of EquiSym and CLIPSym, which take images under random rotation transformations within $[-45°, 45°]$ as inputs to illustrate the model's robustness and consistency. We observe that even though EquiSym is a fully equivariant model, CLIPSym generates more consistent heatmaps. This is because end-to-end equivariant models using steerable filters require exact symmetry at the input. They are not guaranteed to be equivariant when there are interpolation artifacts, cropping of the image, or when the rotation is not a multiple of $90°$. On the contrary, CLIP's image encoder is robust to such transformations due to large-scale training and generates consistent image features. Our equivariant decoder module D, generates a consistent symmetry heatmap from CLIP's image feature.

Fig. 3 also shows that compared with EquiSym, CLIPSym's predictions are sharper and contain less noise, suggesting that CLIPSym is more robust. More results are shown in Fig. A4.

4.3. Ablation studies

Prompt initialization. In Tab. 5, we investigated the impact of different prompt initialization methods for the CLIPSym text encoder on the reflection symmetry detection performance using the DENDI dataset. We explored two main categories of prompt initialization: single prompt ($M = 1$) and multiple prompts ($M > 1$).

For a single prompt, we evaluated using arbitrary phrases or sentences, such as "reflection axis" and "symmetry axis", which achieved F1 scores of 64.4 and 64.8, respectively. We also tested combinations of words with frequent objects (65.3). Furthermore, using multiple prompts containing M tokens each consistently outperforms single prompt methods.

We find that using 25 prompts with 4 tokens each yields the highest F1 score of 66.5, demonstrating the effectiveness of leveraging multiple diverse prompts for initialization. These results highlight the importance of careful prompt engineering and show that utilizing multiple semantically relevant prompts can improve performance. Appendix A1 provides more detailed descriptions of the prompts.

	Prompt context	Ref. F1
Single prompt	*A single phrase containing K tokens*	
	"reflection axis"	64.4
	"symmetry axes in the image"	64.8
	frequent object classes (K=25)	**65.8**
Multi-prompt	*M prompts, each with K tokens*	
	$M = 25, K = 1$	65.3
	$M = 25, K = 4$	**66.5**
	$M = 25, K = 16$	65.9
	$M = 50, K = 4$	65.4
	$M = 50, K = 16$	64.4

Table 5. Ablation results of different prompt initialization methods for CLIPSym text encoder on DENDI reflection dataset. As defined in Sec. 3.2, M represents the number of prompts, and K represents how many words there are in each prompt.

Trainable Encoder		Ref. F1
Text	*Image*	
✗	✗	59.4
✓	✗	58.9
✗	✓	65.3
✓	✓	**66.5**

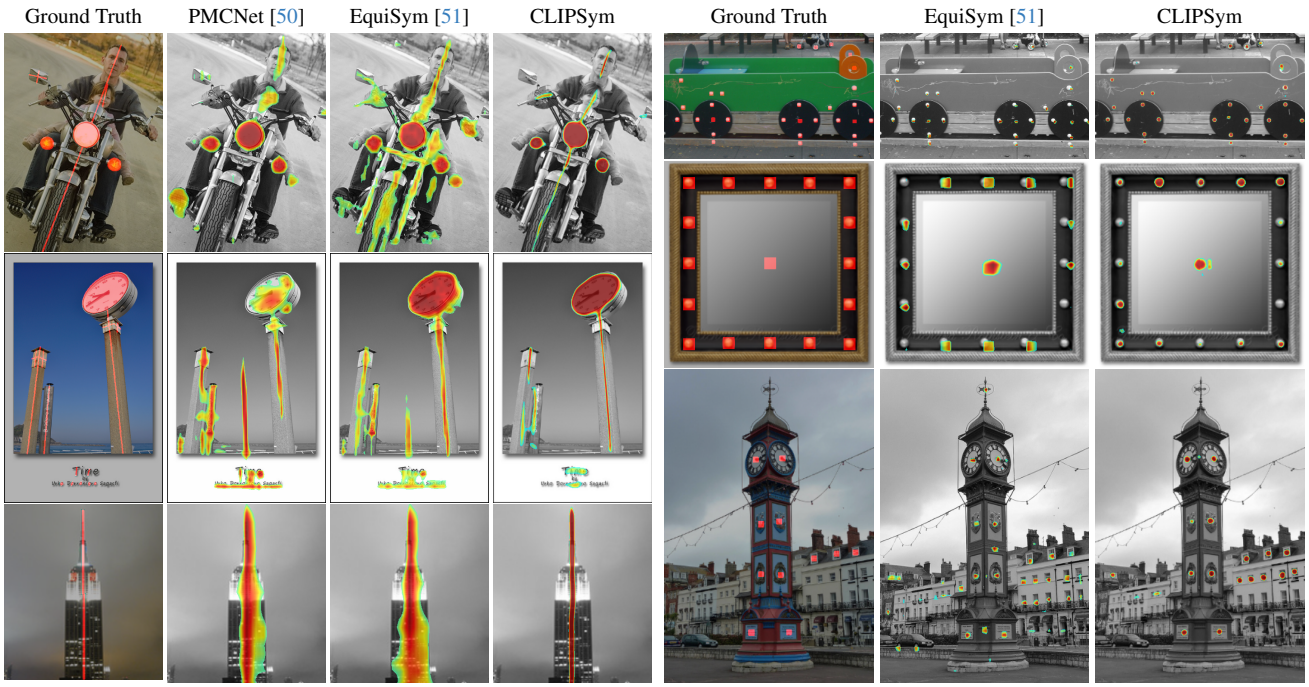
Table 6. F1-scores evaluated on DENDI reflection dataset under different settings.

CLIP Version	Ref. F1
CLIP/ViT-B-16	66.5 ± 0.2
CLIP/ViT-L-14	65.4 ± 0.2
SigLIP/ViT-B-16	65.8 ± 0.3
MetaCLIP/ViT-B-16	$\mathbf{66.7 \pm 0.3}$

Table 7. Comparison of different versions of CLIP model on reflection symmetry detection on DENDI.

Trainable components. In Tab. 6, we investigate the impact of making the text encoder and image encoder trainable in the proposed CLIPSym. The best performance is achieved when both encoders are trainable, which suggests that both encoders contribute to the symmetry detection task. When the image encoder is frozen, whether the text encoder is trainable or not, the performance drops significantly. This suggests that the image encoder plays a more crucial role in symmetry detection than the text encoder.

Different CLIP models. Beyond using the ViT-B/16 model as in the main experiments, we also experimented with other variants of CLIP models, including ViT-L/14, SigLIP [76] which replaces the softmax loss with a sigmoid loss for improved feature separability, and MetaCLIP [67] which created a balanced and noise-reduced dataset to improve the training of CLIP model. As reported in Tab. 7, we observe that MetaCLIP achieves even better reflection detection performance than our model (66.7 vs. 66.5), and SigLIP achieves slightly worse performance (65.8). This suggests that CLIPSym has the potential to be further improved as more advanced CLIP backbones are developed.



(a) Reflection detection results on DENDI-*ref*.

(b) Rotation detection results on DENDI-*rot*.

Figure 2. Visualization of the reflection and rotation symmetry detection on the DENDI dataset.

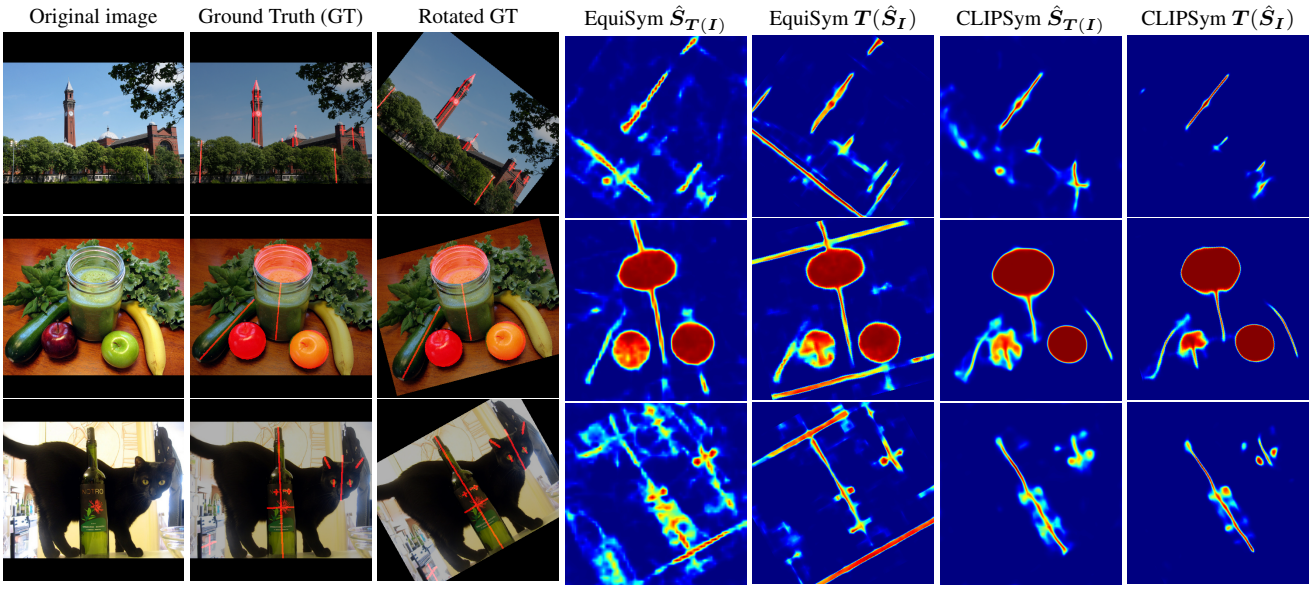


Figure 3. Examples of the original image, ground truth, rotated ground truth, EquiSym and CLIPSym's predicted heatmaps $\hat{S}_{T(I)}$ on the rotated image and the rotated heatmap $T(\hat{S}_I)$. Observe that CLIPSym's results are more consistent under rotation transformations.

5. Conclusion

In this paper, we introduce CLIPSym, a new approach for symmetry detection that builds on the pre-trained CLIP model. Leveraging CLIP's powerful generalization and cross-modal capabilities, our method adapts it specifically for symmetry detection through prompt learning to capture geometrically relevant features. Additionally, the proposed equivariant decoder module boosts the model's robustness and consistency against random transformations. Our approach achieves state-of-the-art performance across all evaluated datasets.

References

[1] I. R. Atadjanov and S. Lee. Reflection symmetry detection via appearance of structure descriptor. In *ECCV*, 2016. 1, 2

[2] H. Bahng, A. Jahanian, S. Sankaranarayanan, and P. Isola. Exploring visual prompts for adapting large-scale models. *arXiv preprint arXiv:2203.17274*, 2022. 2

[3] M. Cicconet, V. Birodkar, M. Lund, M. Werman, and D. Geiger. A convolutional approach to reflection symmetry. *Pattern Recognition Letters*, 2017. 1

[4] T. Cohen and M. Welling. Group equivariant convolutional networks. In *ICML*, 2016. 2, 4

[5] T. S. Cohen and M. Welling. Steerable CNNs. In *ICLR*, 2017. 2, 4

[6] J. D. Delius and G. Habers. Symmetry: can pigeons conceptualize it? *Behavioral biology*, 1978. 1

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3, 17

[8] J. Driver, G. C. Baylis, and R. D. Rafal. Preserved figure-ground segregation and symmetry perception in visual neglect. *Nature*, 1992. 1

[9] V. Dumoulin, E. Perez, N. Schucher, F. Strub, H. d. Vries, A. Courville, and Y. Bengio. Feature-wise transformations. *Distill*, 2018. 3

[10] A. Ecins, C. Fermüller, and Y. Aloimonos. Cluttered scene segmentation using the symmetry constraint. In *ICRA*, 2016. 1

[11] K. Fukushima and M. Kikuchi. Symmetry axis extraction by a neural network. *Neurocomputing*, 2006. 2

[12] C. Funk and Y. Liu. Beyond planar symmetry: Modeling human perception of reflection and rotation symmetries in the wild. In *ICCV*, 2017. 1, 2, 5, 6

[13] C. Funk, S. Lee, M. R. Oswald, S. Tsogkas, W. Shen, A. Cohen, S. Dickinson, and Y. Liu. 2017 iccv challenge: Detecting symmetry in the wild. In *ICCVW*, 2017. 1

[14] A. Gnutti, F. Guerrini, and R. Leonardi. Combining appearance and gradient information for image symmetry detection. *IEEE TIP*, 2021. 2

[15] J. Hartford, D. Graham, K. Leyton-Brown, and S. Ravanbakhsh. Deep models of interactions across sets. In *Proc. ICML*, 2018. 2

[16] D. C. Hauagge and N. Snavely. Image matching using local symmetry features. In *CVPR*, 2012. 1

[17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *ICCV*, 2016. 5

[18] X. Hu, C. Zhang, Y. Zhang, B. Hai, K. Yu, and Z. He. Learning to adapt CLIP for few-shot monocular depth estimation. In *WACV*, 2024. 2

[19] J. Je, J. Liu, G. Yang, B. Deng, S. Cai, G. Wetzstein, O. Litany, and L. Guibas. Robust symmetry detection via riemannian langevin dynamics. In *SIGGRAPH Asia*, 2024. 1

[20] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. Le, Y.-H. Sung, Z. Li, and T. Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. 1

[21] M. Kawano, W. Kumagai, A. Sannai, Y. Iwasawa, and Y. Matsuo. Group equivariant conditional neural processes. *arXiv preprint arXiv:2102.08759*, 2021. 2

[22] Y. Keller and Y. Shkolnisky. A signal processing approach to symmetry detection. *IEEE TIP*, 2006. 2

[23] M. U. Khattak, H. Rasheed, M. Maaz, S. Khan, and F. S. Khan. Maple: Multi-modal prompt learning. In *CVPR*, 2023. 2

[24] S. Lee and Y. Liu. Skewed rotation symmetry group detection. *IEEE TPAMI*, 2009. 2

[25] A. Lengyel, O. Strafforello, R.-J. Bruintjes, A. Gielisse, and J. van Gemert. Color equivariant convolutional networks. In *NeurIPS*, 2024. 2

[26] H.-C. Lin, L.-L. Wang, and S.-N. Yang. Extracting periodicity of a regular texture based on autocorrelation functions. *Pattern recognition letters*, 1997. 2

[27] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 5

[28] I.-J. Liu, R. A. Yeh, and A. G. Schwing. Pic: permutation invariant critic for multi-agent deep reinforcement learning. In *Proc. CORL*, 2020. 2

[29] I.-J. Liu, Z. Ren, R. A. Yeh, and A. G. Schwing. Semantic tracklets: An object-centric representation for visual multi-agent reinforcement learning. In *Proc. IROS*, 2021. 2

[30] J. Liu, G. Slota, G. Zheng, Z. Wu, M. Park, S. Lee, I. Rauschert, and Y. Liu. Symmetry detection from realworld images competition 2013: Summary and results. In *CVPRW*, 2013. 1

[31] Y. Liu, H. Hel-Or, C. S. Kaplan, L. Van Gool, et al. Computational symmetry in computer vision and computer graphics. *Foundations and Trends® in Computer Graphics and Vision*, 2010. 1

[32] T. Lüddecke and A. Ecker. Image segmentation using text and image prompts. In *CVPR*, 2022. 2

[33] M. Lukáč, D. Sỳkora, K. Sunkavalli, E. Shechtman, O. Jamriška, N. Carr, and T. Pajdla. Nautilus: Recovering regional symmetry transformations for image editing. *ACM TOG*, 2017. 1

[34] S. Park, A. Senocak, and J. S. Chung. Can CLIP help sound source localization? In *WACV*, 2024. 2

[35] H. Pashler. Coordinate frame for symmetry detection and object recognition. *Journal of Experimental Psychology: Human Perception and Performance*, 1990. 1

[36] V. S. N. Prasad and L. S. Davis. Detecting rotational symmetries. In *ICCV*, 2005. 1, 2

[37] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1, 2, 6

[38] M. A. Rahman and R. A. Yeh. Truly scale-equivariant deep nets with fourier layers. In *NeurIPS*, 2024. 2

[39] M. A. Rahman and R. A. Yeh. Group downsampling with equivariant anti-aliasing. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=sOte83GogU. 2

[40] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021. 1

[41] H. Rasheed, M. U. Khattak, M. Maaz, S. Khan, and F. S. Khan. Fine-tuned CLIP models are efficient video learners. In *CVPR*, 2023. 2

[42] I. Rauschert, J. Liu, K. Brockelhurst, S. Kashyap, and Y. Liu. Symmetry detection competition: A summary of how the competition is carried out. In *CVPRW*, 2011. 1

[43] S. Ravanbakhsh, J. Schneider, and B. Poczos. Deep learning with sets and point clouds. In *Proc. ICLR workshop*, 2017. 2

[44] T. Ren, S. Liu, A. Zeng, J. Lin, K. Li, H. Cao, J. Chen, X. Huang, Y. Chen, F. Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024. 12

[45] I. Rodríguez, A. Gumbert, N. Hempel de Ibarra, J. Kunze, and M. Giurfa. Symmetry is in the eye of the 'beeholder': innate preference for bilateral symmetry in flower-naïve bumblebees. *Naturwissenschaften*, 2004. 1

[46] R. A. Rojas-Gomez, T.-Y. Lim, A. Schwing, M. Do, and R. A. Yeh. Learnable polyphase sampling for shift invariant and equivariant convolutional networks. *Advances in Neural Information Processing Systems*, 35:35755–35768, 2022. 2

[47] R. A. Rojas-Gomez, T.-Y. Lim, M. N. Do, and R. A. Yeh. Making vision transformers truly shift-equivariant. In *CVPR*, 2024. 2

[48] C. Schuhmann, R. Vencu, R. Beaumont, R. Kaczmarczyk, C. Mullis, A. Katta, T. Coombes, J. Jitsev, and A. Komatsuzaki. Laion-400M: Open dataset of CLIP-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 1, 12

[49] A. Seo and M. Cho. Leveraging 3d geometric priors in 2d rotation symmetry detection. In *CVPR*, 2025. 1, 17

[50] A. Seo, W. Shim, and M. Cho. Learning to discover reflection symmetry via polar matching convolution. In *ICCV*, 2021. 1, 2, 5, 6, 7, 8, 14, 15, 17

[51] A. Seo, B. Kim, S. Kwak, and M. Cho. Reflection and rotation symmetry detection via equivariant learning. In *CVPR*, 2022. 1, 2, 3, 5, 6, 7, 8, 13, 14, 15, 17

[52] D. Shen, H. H. Ip, and E. K. Teoh. Robust detection of skewed symmetries by combining local and semi-local affine invariants. *Pattern Recognition*, 2001. 1, 2

[53] S. N. Sinha, K. Ramnath, and R. Szeliski. Detecting and reconstructing 3d mirror symmetric objects. In *ECCV*, 2012. 1

[54] I. Sosnovik, M. Szmaja, and A. Smeulders. Scale-equivariant steerable networks. *arXiv preprint arXiv:1910.11093*, 2019. 2

[55] S. Tsogkas and I. Kokkinos. Learning-based symmetry detection in natural images. In *ECCV*, 2012. 2

[56] C. W. Tyler. *Human symmetry perception and its computational analysis*. Psychology Press, 2003. 1

[57] T. Vetter, T. Poggio, and H. Bülthoff. The importance of symmetry and virtual views in three-dimensional object recognition. *Current Biology*, 1994. 1

[58] L. von Fersen, C. S. Manos, B. Goldowsky, and H. Roitblat. Dolphin detection and conceptualization of symmetry. *Marine mammal sensory systems*, 1992. 1

[59] Z. Wang, L. Fu, and Y. Li. Unified detection of skewed rotation, reflection and translation symmetries from affine invariant contour features. *Pattern recognition*, 2014. 2

[60] Z. Wang, Z. Tang, and X. Zhang. Reflection symmetry detection using locally affine invariant edge correspondence. *IEEE TIP*, 2015. 2

[61] M. Weiler and G. Cesa. General E(2)-equivariant steerable cnns. In *NeurIPS*, 2019. 2

[62] M. Weiler, M. Geiger, M. Welling, W. Boomsma, and T. S. Cohen. 3D steerable CNNs: Learning rotationally equivariant features in volumetric data. In *NeurIPS*, 2018.

[63] M. Weiler, F. A. Hamprecht, and M. Storath. Learning steerable filters for rotation equivariant cnns. In *CVPR*, 2018. 2

[64] J. Wilder, M. Rezanejad, S. Dickinson, K. Siddiqi, A. Jepson, and D. B. Walther. Local contour symmetry facilitates scene categorization. *Cognition*, 2019. 1

[65] D. Worrall and M. Welling. Deep scale-spaces: Equivariance over scale. In *NeurIPS*, 2019. 2

[66] Z. Wu, Y. Liu, H. Dong, X. Tang, J. Yang, B. Jin, M. Chen, and X. Wei. R2det: Exploring relaxed rotation equivariance in 2d object detection. In *ICLR*, 2025. 1

[67] H. Xu, S. Xie, X. E. Tan, P.-Y. Huang, R. Howes, V. Sharma, S.-W. Li, G. Ghosh, L. Zettlemoyer, and C. Feichtenhofer. Demystifying CLIP data. *arXiv preprint arXiv:2309.16671*, 2023. 7

[68] R. Xu, K. Yang, K. Liu, and F. He. $E(2)$-equivariant vision transformer. In *Proc. UAI*, 2023. 2

[69] C. Xue, W. Zhang, Y. Hao, S. Lu, P. H. Torr, and S. Bai. Language matters: A weakly supervised vision-language pre-training approach for scene text detection and spotting. In *ECCV*, 2022. 2

[70] R. Yeh, Y.-T. Hu, and A. Schwing. Chirality nets for human pose regression. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[71] R. A. Yeh, A. G. Schwing, J. Huang, and K. Murphy. Diverse generation for multi-agent sports games. In *Proc. CVPR*, 2019. 2

[72] R. A. Yeh, Y.-T. Hu, M. Hasegawa-Johnson, and A. Schwing. Equivariance discovery by learned parameter-sharing. In *International Conference on Artificial Intelligence and Statistics*, pages 1527–1545. PMLR, 2022. 2

[73] L. Yuan, D. Chen, Y.-L. Chen, N. Codella, X. Dai, J. Gao, H. Hu, X. Huang, B. Li, C. Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021. 1

[74] C. Yun, S. Bhojanapalli, A. S. Rawat, S. J. Reddi, and S. Kumar. Are transformers universal approximators of sequence-to-sequence functions? In *ICLR*, 2020. 4

[75] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola. Deep sets. In *NeurIPS*, 2017. 2, 4

[76] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer. Sigmoid loss for language image pre-training. In *ICCV*, 2023. 7

[77] K. Zhou, J. Yang, C. C. Loy, and Z. Liu. Conditional prompt learning for vision-language models. In *CVPR*, 2022. 2

[78] K. Zhou, J. Yang, C. C. Loy, and Z. Liu. Learning to prompt for vision-language models. *IJCV*, 2022. 2