Unseen defect image synthesis with compositional conditional diffusion model

Pin-Chuan Chen[1]
barry556652.ai10@nycu.edu.tw
Shih-Lun Lai[1]
larry.ai10@nycu.edu.tw
Ching-Wen Ma[2]
machingwen@nycu.edu.tw

[1] Institute of Computational
Intelligence, NYCU
Tainan, ROC

[2] National Yang Ming
Chiao Tung University
Tainan, ROC

Abstract

This study introduces an approach aimed at enhancing defect detection in PCB manufacturing through the utilization of Compositional Conditional Diffusion Models (CCDM). Focused on tackling the challenge of detecting defects in new component categories with no samples available, CCDM harnesses deep learning techniques to generate images showcasing specific defect features. Drawing insights from developments ranging from non-equilibrium thermodynamics [21] to Denoising Diffusion Probabilistic Models [10], the study underscores the effectiveness of CCDM in efficiently producing high-quality defect images for new components. This approach significantly reduces the human and computational resources required for retraining models for each new category.

1 Introduction

This research explores advancing image generation techniques to effectively create defect features for new component categories with limited or no defect data. By focusing on overcoming traditional models' shortcomings in generating realistic defects for new components, we aim to develop a more efficient model structure. This model leverages Diffusion model, statistical methods, and training data from old components to generate high-quality, diverse defect images, addressing the challenge of insufficient defect samples for new components and pushing the boundaries of current image generation capabilities.

Taiwan's leading PCB industry faces challenges with defect detection in new components, impacting yield rates and reputations. Traditional AI defect detection systems struggle with these unseen components, highlighting the need for advanced image generation technology. The PCB process, prone to defects due to high temperatures and pressures, underscores the importance of accurately generating images of new component defects. While text-to-image techniques like OpenAI's Stable Diffusion [18] have limitations in industrial applications, Conditional Diffusion Models (CDMs) [2, 3, 23, 30]show promise by generating specific defect images based on varied conditions. CDMs can effectively learn from existing data to produce a wider range of realistic defect scenarios, improving defect detection accuracy and maintaining manufacturing quality and efficiency.

This study utilizes a PCB dataset from a collaborative industry partner to explore the generation of "zero-shot" [25] images and enhance defect detection. The dataset, as shown in Figure 1, we present the representation of component groups and defect types,initially used for defect detection research, will now also support generating images for missing component groups, specifically "Broke Group3," using a Compositional Class to image approach, Classifier-Free Diffusion Guidance [9] and Denoising Diffusion Implicit Models (DDIM) [22] for faster generation. This method aims to improve the model's ability to detect defects in unseen components, thereby contributing to the field's advancement and providing inspiration for future research.

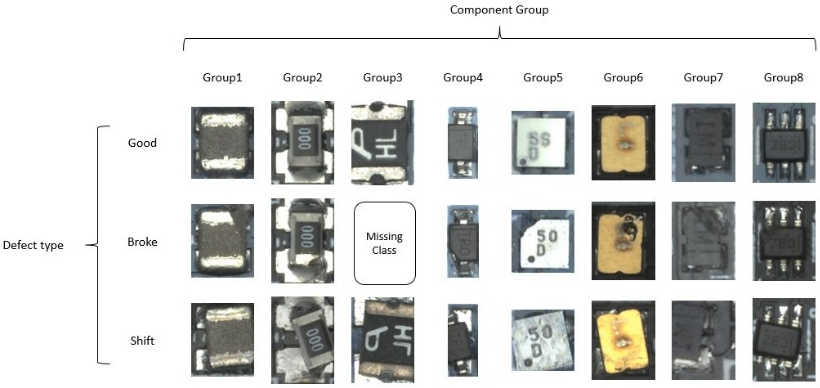

Figure 1: The representation of component groups and defect types

2 Related works

Compositional Zero-Shot Learning (CZSL) [12, 13, 14, 20, 29, 31] extends **Zero-Shot Learning (ZSL)** [1, 6, 15, 17, 25, 27, 28] by teaching models to recognize new combinations of unseen classes, adding complexity to the conventional ZSL approach that enables models to identify categories unseen during training through learned category relationships. Unlike ZSL, which predicts and compares attributes, CZSL directly trains models on recognizing new category compositions without the need for attribute prediction, allowing for a more straightforward generalization to novel categories. This methodology has applications across various fields, including natural language processing, computer vision, and multimodal data tasks, addressing advanced zero-shot learning challenges.

Conditional Diffusion Models represent a significant advancement in image generation, combining diffusion processes with the precision of conditional generation. Unlike traditional diffusion models [10] focused on unsupervised generation, these models use conditional information, like class labels or textual descriptions, to guide the image creation process. This approach enables the generation of images based on specific conditions, providing precise control over the output. Such capability is particularly useful in applications requiring detailed customization, such as defect synthesis in manufacturing. Conditional Diffusion Models [2, 4, 23, 30], are mathematically defined to generate images by incorporating conditioning data, enhancing their ability to produce images with targeted visual characteristics. This makes them a valuable tool in the realm of advanced image generation, especially for tasks needing fine-grained control over generated content. In simpler terms, recent advancements in iterative generative models like DDPMs [10] and score-based generative models have achieved image generation quality on par with GANs [5]. Among these, Denoising Diffusion Implicit Models (DDIMs) [22] stand out by producing high-quality images in fewer steps. Specifically, enhancements to the DDIM architecture have led to state-of-the-art performance in image synthesis [24]. A key advantage of diffusion models over GANs is their non-reliance on adversarial training, effectively addressing the issue of mode collapse that GANs face.Variational Autoencoders (VAEs) [11] and flow-based models achieve an efficient synthesis of high-resolution images, yet their sample quality falls short of GANs.In the end,We chose to use a Diffusion model as our model for generating images.

3 Compositional Condition Diffusion Model

In this chapter, a methodology employing the Compositional Condition Diffusion Model (CCDM) is proposed, exploring its application in generating defects for new components. Initially, an extensive dataset from industrial production is utilized, categorized into normal and defective samples, and further classified based on different component types. Subsequently, the model is trained through four key stages.

3.1 Model Architectures

Denoising Diffusion Probabilistic Models (DDPMs) [10] are advanced generative models used for image generation, transforming simple distributions into complex data distributions through a learned Markov chain gradually adds noise to latent variables $x_1,...,x_T$ sampled sequentially from the same dimensions. The generative process involves gradually adding noise to latent variables, with the objective of transforming an initial sample x_0 under a compositional condition c, characterized by a combination of attribute and object embeddings. Denoted as c is concatenated from c_{obj} and c_{atr}. This labeling approach signifies the integration of these two embeddings.Here, each step in the forward process is a Gaussian translation.

$$q(x_t|x_{t-1},c) := \mathcal{N}(x_t\,|\,c; \sqrt{1-\beta_t}x_{t-1}\,|\,c, \beta_t I) \tag{1}$$

In this process, a fixed schedule of variances, denoted as $\beta_1,\ldots,\beta_T$, is utilized instead of learned parameters. The procedure involves obtaining x_t by introducing a small Gaussian noise to the latent variable. Given an initial clean data point x_0, the sampling of x_t can be explicitly expressed in a closed form.

$$q(x_t|x_0,c) := \mathcal{N}(x_t\,|\,c; \sqrt{\bar{\alpha}_t}x_0\,|\,c, (1-\bar{\alpha}_t)I) \tag{2}$$

where α_t:= 1 - β_t and $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$. Then a conditional U-Net [19] $\varepsilon_\theta(x,t,c)$ is trained to approximate the reverse denoising process,

$$p_\theta(x_{t-1}|x_t,c) := \mathcal{N}(x_{t-1}\,; \mu_\theta(x_t,t,c); \Sigma_\theta(x_t,t,c)) \tag{3}$$

The variance μ_σ can be learnable parameters or a fixed set of scalars. As for the mean, after reparameterization with $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon$ for $\varepsilon \sim \mathcal{N}(0,I)$, the loss function can be simplified as:

$$L := E_{x_0,\varepsilon}\,||\,\varepsilon - \varepsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon, t, c)\,|| \tag{4}$$

To incorporate the binary condition $c$into the U-Net architecture, we adopt a strategy inspired by. This involves employing an embedding projection function, denoted as $e = f(c)$, where $f \in R \rightarrow R^n$, and n represents the embedding dimension. Subsequently, the condition embedding is added to feature maps across every Resblocks [8]. Following the training of the denoising model, empirical evidence demonstrates that the network is capable of generating the desired conditional distribution $D(x|c)$ given the compositional condition c.

Our objective is to derive a segmentation mask from samples generated through a few reverse Markov steps using DDIM . The rationale behind choosing DDIM lies in its capability to deterministically generate a sample x_{t-1} from x_t by eliminating the random noise term.

$$x_{t-1}(x_t,t,c) = \sqrt{\bar{\alpha}_{t-1}}\frac{(x_t - \sqrt{1-\bar{\alpha}_t}\hat{\varepsilon}(x_t,c)}{\sqrt{\bar{\alpha}_t}} + \sqrt{1-\bar{\alpha}_{t-1}}\hat{\varepsilon_\theta}(x_t,c) \tag{5}$$

The forward diffusion process is modeled as Gaussian transitions with a fixed variance schedule, allowing for the explicit sampling of latent variables at any step. A conditional U-Net is then trained to reverse this process, aiming to denoise and generate the desired image by predicting the noise. The loss function focuses on minimizing the difference between the actual and predicted noise, with the condition c integrated into the U-Net through embedding projection. This framework is particularly effective in generating images under specific conditions, and is further refined using Denoising Diffusion Implicit Models (DDIMs) for deterministic sampling and segmentation mask derivation, emphasizing DDIMs' ability to generate samples with reduced randomness in the reverse Markov steps, as illustrated in Figure 2.

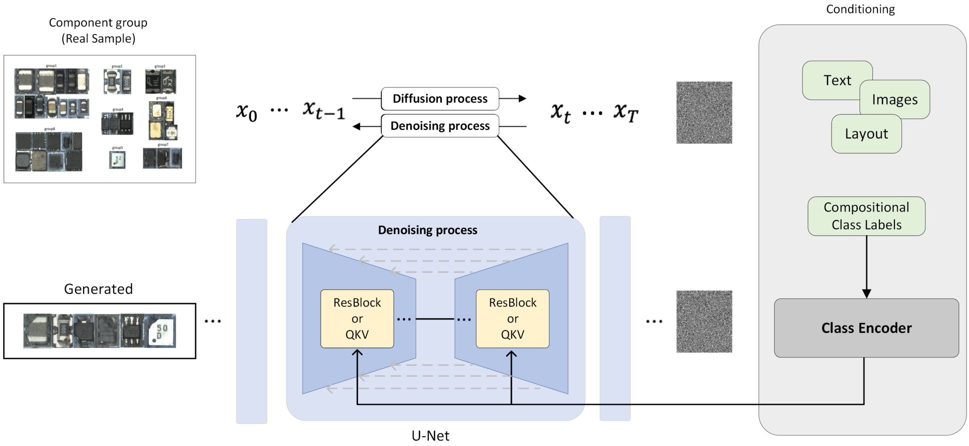

Figure 2: Compositional Conditional Diffusion model

3.2 Conditioning Mechanisms

The widely recognized diffusion models the Stable Diffusion Model, known for its effectiveness in general scenarios. However, our focus extends beyond generating images commonly found in everyday life; we target electronic components used in industrial settings. Additionally, our research is to systematically synthesize images of electronic components based on their compositional the application of advanced generative models in the realm of industrial image synthesis.

Furthermore, we aim to pioneer exploration in the underdeveloped domain of Compositional Class-to-Image approaches. To realize this, we adopt a Class Encoding method for encoding the attributes of electronic components (e.g., "Good," "Broke," etc.) and the group names of electronic components.

Object conditions(Groups) : Object conditions refer to specifications that dictate the content of generated images, representing the appearance or category of the desired objects. While CLIP (Contrastive Language-Image Pre-Training) [16] has been widely employed in natural language processing, it faces limitations in handling out-of-vocabulary (OOV) words in the context of text-to-image generation for industrial electronic components. To address this challenge, we draw inspiration from the Stable Diffusion model and make refinements in encoding object conditions. The generation process for object conditions can be described as follows:

$$c_{obj} = \mathrm{Proj}(\mathrm{Emb}(\mathrm{Encoder}(obj))) \tag{6}$$

Attribute conditions(Defect type) : Specifically designed for the Defect Type in electronic components, it captures the unique appearance and characteristics of specific component damages. In simpler terms, the Attribute condition reflects the Defect Type of the component. The generation process of the Attribute Condition can be expressed as:

$$c_{atr} = \mathrm{Proj}(\mathrm{Emb}(\mathrm{Encoder}(atr))) \tag{7}$$

These two encoded representations are subsequently embedded, as depicted in Figure 3.3. Following this, the embedded vectors are concatenated to yield a unified embedding. Lastly, utilizing an condition mechanism, this combined embedding is mapped to the ResBlock and integrated with timesteps within the U-Net.

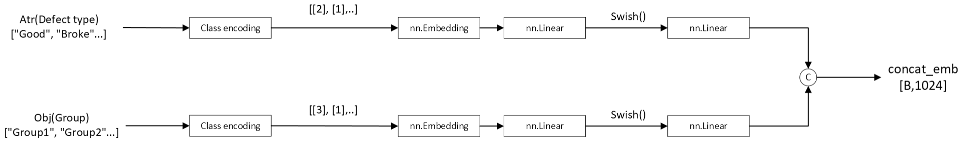

Figure 3: Condition Module

3.3 U-Net

The U-Net architecture, initially designed for biomedical image segmentation, has emerged as a pivotal framework in the field of deep learning for various image processing tasks. At its core, the U-Net architecture is distinguished by its U-shaped design, which comprises a contracting path to capture context and a symmetric expanding path that enables precise localization.The U-Net architecture we employed is based on the structure of Stable Diffusion. We also experiment with a layer that we refer to as addition group normalization (AddGN) [26], which incorporates the timestep and class embedding into each residual block after a group normalization operation. We define this layer as $\mathrm{AddGN}(h,y) = y_s + \mathrm{GroupNorm}(h)$, where h is the intermediate activations of the residual block following the first convolution, and $y = [y_s]$ is obtained from a linear projection of the timestep and class embedding.

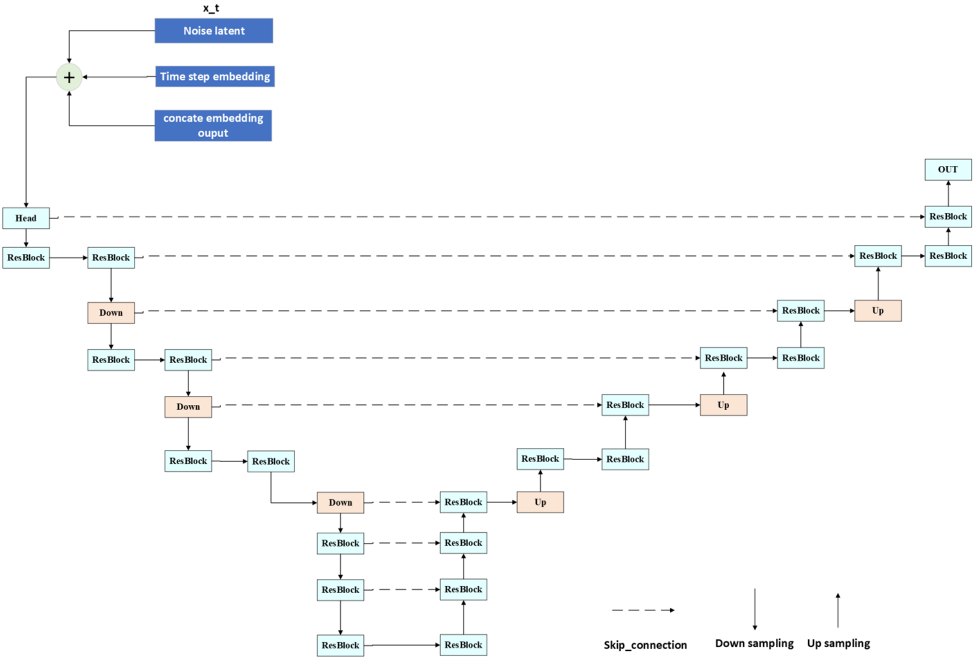

Figure 4: Unet Architecture

3.4 Select image

By combining the frameworks mentioned earlier, we can generate relatively indistinguishable unseen images. After producing 50 unseen images (Broke Group3), it was observed that the accuracy of the generated images is not consistently high, with some exhibiting variations in conditions. Consequently, we designed a simple binary classifier using ResNet18 [7] as the backbone to distinguish between "defective" and "non-defective" images. Afterwords, the generated images are fed into this binary classifier for categorization. More precisely, images generated by CCDM are separated, and the classifier is applied to identify "defective unseen images" chosen through this process. This methodology aims to improve the accuracy of generated images and contributes to enhancing the overall effectiveness of the specify what approach in generating realistic and defect-specific unseen images.

4 Experiments

4.1 Datasets

This research utilizes a PCB defect dataset generously provided by an electronics manufacturing company, encompassing a vast collection of 1,364,400 images. Each image is intricately associated with a distinct defect category and the corresponding component name. The intricate nature of component PCB processes introduces a variety of potential defects, thereby prompting the categorization into six defect types: Good (indicating normalcy), Missing, Shift, Stand, Broken, and Short, as shown in Figure 1.

It is important to note that the "Good" category contains a substantial number of component images, creating a noticeable imbalance compared to the relatively no samples found in other defect categories-particularly within the realm of new components. This imbalance, notably in the new component subset, is a pivotal challenge in our dataset.

To mitigate this challenge, an over-sampling strategy has been employed on the original dataset. Given the scarcity of defect samples, the data's inherent imbalance is addressed by repeatedly sampling from the underrepresented class. This method involves the random duplication of instances, effectively augmenting the less abundant class to achieve a more equitable distribution of data between the two classes.

Additionally, the dataset was meticulously reorganized. Despite variations in nomenclature, certain components exhibit only subtle visual disparities due to the distinctive resistance characteristics of each component. To prevent potential overfitting during the training phase, visually similar components were carefully grouped into the same category or cluster, forming cohesive and homogenous groups.

4.2 Implementation Details

To refine our model, we utilized the Stable-diffusion-2-base pre-trained model on the LAION-5B dataset as a foundational framework, implementing PyTorch with default hyperparameters. The training epochs were set to 100, and the initial learning rate was configured as 5e-5. For optimization, we employed the CosineAnnealingLR, with training commencing after a warm-up scheduler for epochs divide by 10.As shown in Table 1, is our CCDM parameter settings.

Given the diverse sizes of components, all input images were resized uniformly to 64x64 color images. This standardized approach accommodated the inherent variability in component dimensions.To augment our model's training set, random horizontal flip and random vertical flip techniques were employed, providing additional diversity for robust training. Notably, these augmentation strategies enhanced the model's ability to generalize across various component orientations.In contrast, the validation and test sets were maintained in their original form without applying any image augmentation. This ensured a thorough evaluation of the model's performance on previously unseen data, thereby reflecting real-world scenarios.All experiments were conducted using two high-performance NVIDIA Tesla V100 GPU, to optimize computational efficiency.

During the PCB dataset image sampling process, we observed that CCDM could generate images correctly, albeit with a high error rate. Some generated images exhibited discrepancies with the specified compositional conditions. In Figure 5, we sampled 50 images of unseen images (broke group 3), revealing a relatively low accuracy.

4.3 Results

Subsequently, we employed a simple binary classifier to train on the original PCB dataset, enabling the classifier to distinguish between broke and unbroke instances. This approach aimed to refine the selection of accurate images by leveraging the binary classifier's ability to discern the presence of defects. Due to the lack of actual Broke Group3 samples, we conducted an experiment using ResNet152 [7] as the backbone for our classifier, training it on the PCB Dataset categorized into Good, Broken, and Shifted. Initially, without incorporating Broke Group1, the classifier's accuracy in correctly identifying real instances of Broke Group1 was approximately 22%. Subsequently, we included Broke Group1 instances

Compositional diffusion model	
Epoch	100
Batch size	64
Learning rate	5e-5
Optimizer	AdamW
Diffusion steps	1000
Noise Schedule	linear
Channel	128
Depth	2
Channel Multiplier	[1, 2, 2, 2]
Head Channels	4
Dropout	0.15
Weight Decay	1e-4
Embedding Dimension	512

Table 1: The parameter settings of CCDM in PCB Dataset

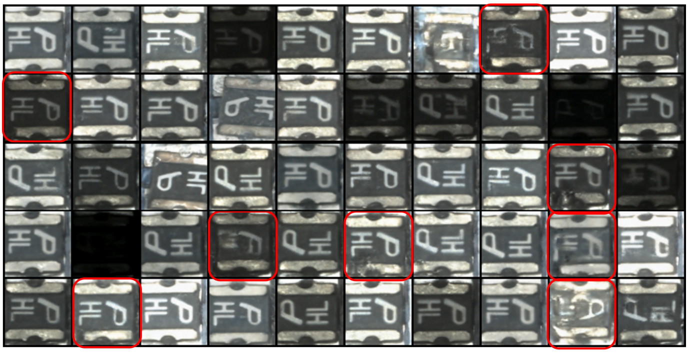

Figure 5: The highlighted portion in the red box represents the results we consider to be correct

generated by CCDM into the classifier's training regimen. This integration led to a significant improvement, enabling the classifier to achieve around 75% accuracy in identifying authentic Broke Group1 instances. This result demonstrates our success in generating new component defect images, which serves as an expansion of the dataset for defect detection applications.Figure 6 is generated by selecting an image produced by CCDM, specifically from broke Group3, we can observe that CCDM exhibit commendable performance in generating unseen images on the PCB dataset.

5 Conclusion

This paper introduces a compositional condition diffusion model designed to generate images of unseen defective components, leveraging class-image correlation for image generation. This approach has practical applications in electronic industry production lines for defect detection, enabling the generation of visual representations for new defective compo-

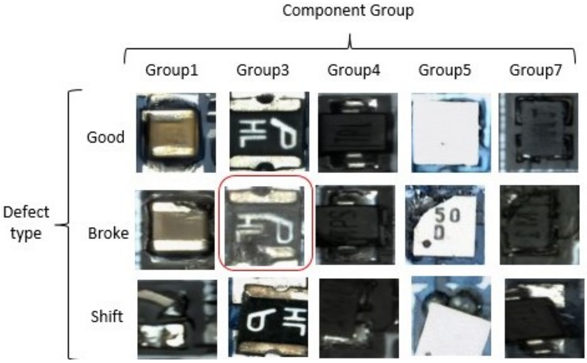

Figure 6: Photos generated by CCDM on the PCB dataset.The part framed in red is an unseen image.

nents. These images can enhance defect detection models through additional training. The research also explores human composite cognitive abilities, focusing on applying learned composite concepts to novel situations, specifically aiming for "composite zero-shot image generation and selection." This study contributes to understanding the generalization capabilities of neural networks in composite zero-shot learning and generation, with aspirations to apply the compositional condition diffusion model across various settings and data types.

References

[1] Wei-Lun Chao, Soravit Changpinyo, Boqing Gong, and Fei Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. pages 52–68, 2016.

[2] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[3] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[4] William Feller. Retracted chapter: On the theory of stochastic processes, with particular reference to applications. pages 769–798, 2015.

[5] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[6] Zongyan Han, Zhenyong Fu, Shuo Chen, and Jian Yang. Contrastive embedding for generalized zero-shot learning. pages 2371–2381, 2021.

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

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. pages 770–778, 2016.

[9] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[10] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[11] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[12] Xiaocheng Lu, Song Guo, Ziming Liu, and Jingcai Guo. Decomposed soft prompt guided fusion enhancing for compositional zero-shot learning. pages 23560–23569, 2023.

[13] Massimiliano Mancini, Muhammad Ferjad Naeem, Yongqin Xian, and Zeynep Akata. Open world compositional zero-shot learning. pages 5222–5230, 2021.

[14] Aditya Panda and Dipti Prasad Mukherjee. Compositional zero-shot learning using multi-branch graph convolution and cross-layer knowledge sharing. *Pattern Recognition*, 145:109916, 2024.

[15] Farhad Pourpanah, Moloud Abdar, Yuxuan Luo, Xinlei Zhou, Ran Wang, Chee Peng Lim, Xi-Zhao Wang, and QM Jonathan Wu. A review of generalized zero-shot learning methods. *IEEE transactions on pattern analysis and machine intelligence*, 2022.

[16] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. pages 8748–8763, 2021.

[17] Shafin Rahman, Salman Khan, and Fatih Porikli. A unified approach for conventional zero-shot, generalized zero-shot, and few-shot learning. *IEEE Transactions on Image Processing*, 27(11):5652–5667, 2018.

[18] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. pages 10684–10695, 2022.

[19] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. pages 234–241, 2015.

[20] Nirat Saini, Khoi Pham, and Abhinav Shrivastava. Disentangling visual embeddings for attributes and objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13658–13667, 2022.

[21] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. pages 2256–2265, 2015.

[22] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[23] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[24] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[25] Wei Wang, Vincent W Zheng, Han Yu, and Chunyan Miao. A survey of zero-shot learning: Settings, methods, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–37, 2019.

[26] Yuxin Wu and Kaiming He. Group normalization. pages 3–19, 2018.

[27] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learninga comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2251–2265, 2018.

[28] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. pages 5542–5551, 2018.

[29] Guangyue Xu, Parisa Kordjamshidi, and Joyce Y Chai. Zero-shot compositional concept learning. *arXiv preprint arXiv:2107.05176*, 2021.

[30] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*, 56(4):1–39, 2023.

[31] Muli Yang, Chenghao Xu, Aming Wu, and Cheng Deng. A decomposable causal view of compositional zero-shot learning. *IEEE Transactions on Multimedia*, 2022.