國立陽明交通大學
智慧計算與科技研究所
碩士論文

Institute of Computational Intelligence
National Yang Ming Chiao Tung University
Master Thesis

基於條件擴散模型的未見缺陷圖像合成

Unseen defect image synthesis with compositional conditional diffusion model

研究生 : 陳品銓 (Pin-Chuan Chen)
指導教授 : 馬清文 (Ching-Wen Ma)

中華民國 一一三年五月
May 2024

基於條件擴散模型的未見缺陷圖像合成

Unseen defect image synthesis with compositional conditional diffusion model

研 究 生：陳品銓
指導教授：馬清文

Student: Pin-Chuan Chen
Advisor: Dr.Ching-Wen Ma

國立陽明交通大學
智慧計算與科技研究所
碩士論文

A Thesis
Submitted to Institute of Computational Intelligence
College of Artificial Intelligence
National Yang Ming Chiao Tung University
in partial Fulfilment of the Requirements
for the Degree of
Master
in
Artificial Intelligence

May 2024

Taiwan, Republic of China

中華民國 一一三年五月

Acknowledgement

畢業論文完成之際，也標誌著我學生生涯的一個階段暫時告一段落。回顧這幾年的碩士生涯，從意外進入交大 AI 學院、涉足電腦視覺領域、日夜加強程式設計技能、獨立承擔 Phison 產學計畫，到最終完成論文，這一路走來，我感激許多人的指導和幫助。

首先，我必須感謝我的指導教授，馬清文教授。馬教授對研究的要求極爲嚴格，常教導我們要追求完美。每當我遇到研究瓶頸或困難時，教授總會耐心討論、提供實用建議並鼓勵我不要放棄。合理的要求叫訓練，不合理的要求叫磨練，謝謝馬老師三年來的磨練。正如老師所言，這段旅程不僅拓寬了我的視野，也讓我對人生有了全新的理解。

我也要感謝實驗室的同學，但大家都樂於互助，尤其要特別感謝世倫。我本科來自護理背景，對資訊工程知識有限，是世倫快速地帶我進入狀況，成爲我碩士生涯中的 AI 導師。同時，感謝伯宏在這段學習旅程中的陪伴。

在 EDA 實驗室，我認識了三位學長：弘運、博群與宙澄。特別是弘運學長的鼓勵，使我能夠一步步完成碩士論文；他的經驗讓我能快速完成寫作。他們畢業後，弘運、博群與宙澄學長也常關心我的狀況，讓我感受到許多溫暖。

最後，我要感謝我的母親。作爲單親家庭的母親，撫養我成長肯定充滿挑戰。儘管我並非一個讓你完全無憂的孩子，但你對我的學業與未來始終給予無條件的支持。當聽聞我考入交大 AI 學院時，你爲我感到高興並全力支持，讓我無需擔心學業與經濟上的壓力。感謝你無私的栽培，讓我能達到今天的成就。

基於條件擴散模型的未見缺陷圖像合成

學生：陳品銓　　指導教授：馬清文 教授

國立陽明交通大學 智慧計算與科技研究所

摘　　要

本研究探討了圖像生成領域中的一項重要問題，即在僅擁有舊元件瑕疵樣本的情況下，如何設計一種能夠有效生成瑕疵特徵的模型結構。焊接作爲一種廣泛應用於工業製造的技術，由於高溫、高壓和可能引發化學反應的特點，容易產生缺陷，進而導致製造過程中斷或最終產品品質不佳。工業圖像異常檢測方面已得到廣泛應用。然而，當處理對新類別的圖像檢測時，這些模型通常難以達到滿足工業生產需求的準確度水平。傳統的重新訓練模型方法，以適應每個新類別的樣本，帶來了巨大的人力和計算成本，使得新類別樣本的檢測成爲該領域面臨的一個重大挑戰。爲解決這一問題，我們提出了一種基於 Conditional diffusion model 的方法，該模型能夠生成具有特定特徵的圖像，同時彌補新元件瑕疵樣本不足的問題。

在背景說明和相關研究部分，我們回顧了生成模型的發展歷程，從 Non-equilibrium thermodynamics 到 Denoising Diffusion Probabilistic Models（DDPM），最終聚焦於 Conditional Diffusion Model。這種模型的特點在於能夠根據不同條件生成具有特定特徵的圖像，並在處理新元件瑕疵樣本不足方面具有優勢。

在研究方法及步驟中，我們提出了一種使用 Compositional Conditional Diffusion Model 的方法，並詳細描述了模型的訓練過程，包括元件標籤、Embedding 生成、Spatial Transformer 的應用等。實驗結果顯示，該模型成功生成了未見過的瑕疵新元件，並進行了模型的優化和改進。

關鍵詞: 深度學習、擴散模型、瑕疵檢測、圖像生成、組合零樣本學習。

Unseen defect image synthesis with compositional conditional diffusion model

Student: Pin-Chuan Chen Advisor: Dr. Ching-Wen Ma

Institute of Computational Intelligence
National Yang Ming Chiao Tung University

Abstract

This study explores a critial issue in the field of image generation: designing a model structure capable of effectively generating defect features when only old component defect samples are available. Welding, as a widely used industrial manufacturing technique, is prone to defects due to its characteristics of high temperature, high pressure, and potential chemical reactions, which can lead to interruptions in the manufacturing process or poor final product quality. Deep learning methods have been widely employed in industrial image anomaly detection. However, when it comes to detecting images of new categories, these models often struggle to achieve the required accuracy levels for industrial production. The conventional approach of retraining models to adapt to samples from each new category entails significant human and computational costs, posing a significant challenge in detecting samples from new categories. To address this problem, we propose an approach based on the Conditional Diffusion Model-a model capable of generating images with specific features while overcoming the challenge of limited defect samples for new components.

In the background and related works section, we review the development of generative models, starting from non-equilibrium thermodynamics to Denoising Diffusion Probabilistic Models (DDPM), ultimately focusing on the Conditional Diffusion Model. CCDM advantage lies in its ability to generate images with specific features based on different conditions, addressing the scarcity of defect samples for new components.

In the research methodology and procedures, we propose a method using the Compositional Conditional Diffusion Model, providing a detailed description of the model's training process, including component labels, embedding generation, and the application of spatial transformers. Experimental results demonstrate the successful generation of previously unseen defect samples

for new components, along with optimizations and improvements to the model.

Keywords: Deep Learning, Diffusion Model, Defect Detection, Image Generation, Compositional Zero-Shot Learning (CZSL)

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Image generation is a process that involves utilizing computer algorithms and models to create new images. This intricate process typically incorporates mathematical and statistical methods, along with extensive training data, enabling computers to generate images with diverse visual features and structures. By leveraging existing data and models, the computational system can produce highly realistic images.

In practical applications, components can be categorized into ”old components,” which have ample normal and defective samples, and ”new components,” where defective samples may be insufficient or lacking. It is crucial to note that the types of defects for both new and old components are identical, as detailed in Table 1.1.

	Normal	Defect
New component	Numerous	Numerous
Old componet	Numerous	Several or None

Table 1.1: The types of defects for both new and old components

Traditional models exhibit suboptimal performance in generating defects for new components. Consequently, our research focuses on designing a more effective model structure for generating defect features when only defective samples from old components are available.

The challenge lies in innovatively addressing the deficiency of defect samples for new components, thereby enhancing the model’s ability to generate realistic and diverse defect features. Through this research, we aim to contribute to the advancement of image generation techniques, particularly in scenarios where limited or no defect data is available for new components.

1.2 Motivation

Taiwan's Printed Circuit Board (PCB) industry holds the leading position in global market share. For PCB manufacturers, the yield rate of circuit boards is crucial; a poor yield rate not only increases costs but also damages corporate reputation. Some manufacturers employ Artificial Intelligence (AI) to develop "Defect Detection" systems, significantly enhancing product quality and inspection efficiency. However, these systems perform poorly in identifying defect types in unseen components, necessitating the use of image generation technology.

Welding is a common technique in industrial manufacturing, involving the connection of electronic components. However, the welding process introduces challenges such as high temperatures, high pressures, and the potential for chemical reactions, making it susceptible to defects. These defects can disrupt the entire manufacturing process or result in subpar quality of the final product. Therefore, the development of imaging techniques capable of generating images of defects in new components becomes crucial.

In the current field of image generation, text-to-image techniques, such as OpenAI's Stable Diffusion [1] method, are widely applied. However, in our experiments, the goal is not to generate generalized patterns but rather specific images related to industrial manufacturing. When providing descriptions of our industrial components to ChatGPT, converting them into prompts, and subsequently utilizing Stable Diffusion for image generation, we observed significant discrepancies and even errors in the generated images.

In addressing these challenges, the use of Conditional Diffusion Models(CDMs) demonstrates significant potential, particularly in handling complex variation processes. The distinctive feature of CDMs lies in their ability to generate images with specific features based on different conditions. They not only model common defect characteristics but also generate corresponding defect images based on the characteristics of different components.

The use of CDMs helps overcome the deficiency of defect samples for new components. By learning from existing data, these models can generate a broader range of defect scenarios based on class conditions, to enhance the accuracy of generating defect images. This, in turn, con-

tributes to minimizing disruptions in the entire manufacturing process and preventing a decline in product quality.

1.3 Goal

The PCB dataset used in this study was provided by a collaborative industry partner of our laboratory, and was used previously for defect detection research. In the initial phase of this project, the dataset was used to investigate the generation of ”zero-shot” images, and it will be employed as an augmented dataset for defect detection in the subsequent stages of this research.

As shown in Figure 1.1, we present the representation of component groups and defect types. It can be observed that the combination within the ”Broke Group3” is not present in this dataset.

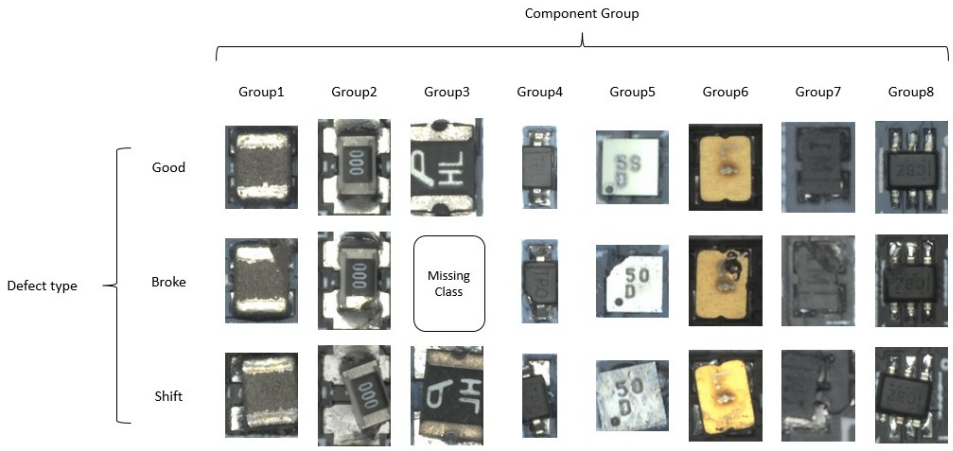

Figure 1.1: The representation of component groups and defect types

This study aims to employ an estimation method based on diffusion processes, utilizing a Compositional Class-to-image approach to generate defect images for new components (Broke Group3). The produced new components (Broke Group3) will be integrated into the defect component detection model to enhance the model’s generalization ability to unseen images. Additionally, we implement Denoising Diffusion Implicit Models(DDIM) [2] sampling to ex-

pedite the generation process. The overarching goal is to provide direction and inspiration for future research in this domain.

1.4 Contribution

The contributions of this thesis are summarized in the following:

- We propose a novel class-to-image generation method that reduces training time by eliminating the need for additional text pre-training models, such as CLIP [3].
- Unlike traditional prompts requiring lengthy descriptions, our method does not rely on complex textual inputs yet is capable of producing accurate and exceptional images even in unseen contexts. This approach simplifies the generation process while ensuring both accuracy and diversity in the generated images.
- **Novelty in New Component Defect Generation Model**: Our study pioneers the use of the Compositional Conditional Diffusion method for generating defects in previously unseen new components. This groundbreaking approach represents a new contribution to the field, introducing a unique and innovative model for defect generation.
- **Practical Application Scenarios**: The potential real-world impact of our model, especially in the industrial welding domain, underscores its practical utility. This application-oriented contribution enhances the relevance and applicability of our research in addressing challenges within the welding industry.

Chapter 2

Related Works

2.1 Image Classification

In the landscape of defect analysis and generation, image classification plays a crucial role in identifying and categorizing various types of defects. This process involves training models to recognize specific patterns, features, and anomalies within images, contributing to the broader understanding of defects in components.

Image classification models are instrumental in distinguishing between normal and defective components based on visual cues. These models leverage machine learning techniques, often utilizing convolutional neural networks (CNNs) [4] or other deep learning architectures to extract and learn intricate features from images.

A key aspect of image classification in defect analysis is it ability to categorize defects into different classes, facilitating a systematic approach to defect identification. This categorization is pivotal for generating accurate defect images and contributes to the overall effectiveness of defect analysis models.

In the context of defect generation, image classification serves as a foundational step in providing necessary labels and conditions for the generative model. Accurate classification of defects in the training data enhances the generative model's ability to simulate realistic defects in new components during the generation process. This synergy between defect classification and generation is vital for improving the accuracy and diversity of generated defect images.

It is noteworthy that our dataset exhibits a significant imbalance, with some defect components having very few samples, or even none. This data imbalance poses a challenge during the training of image classification models, as the model may tend to focus more on categories with more samples, leading to suboptimal learning for minority categories. This further emphasizes

the complexity and uniqueness of our work.

2.2 Compositional zero-shot learning(CZSL)

Compositional Zero-Shot Learning (CZSL) [5–10] is an extension of Zero-Shot Learning (ZSL) [11–17]. In traditional ZSL, models are required to recognize classes that were not shown during training. However, CZSL adds a layer of complexity by demanding that the model handle combinations of previously unseen classes.

2.2.1 Zero-Shot learning

In ZSL, the objective is to enable models to recognize categories that were not encountered during the training process. This is achieved by learning some form of relationship between categories. For example, a model might learn that ”all cats have ears, eyes, tails, etc.,” even if it has not seen a specific breed of cat during training. The prediction process in ZSL typically involves two steps: first, predicting the attributes (atr) of the target category based on available information (such as textual descriptions); second, determining the most similar category by comparing these predicted attributes with those of known categories.

2.2.2 Compositional Zero-shot learning

Unlike ZSL, CZSL does not rely on a mechanism of predicting attributes followed by comparison. The goal of CZSL is to directly teach the model to recognize new category compositions from unseen categories (for example, by observing different objects and their parts), thereby enabling it to identify new category combinations. This approach focuses more on how known concepts (such as shapes, colors, parts, etc.) can be combined to recognize new objects, rather than first predicting the attributes of these objects. Thus, CZSL can generalize to new categories more directly, without the need for explicit definition and prediction of attributes.

Applications of compositional zero-shot learning span various domains such as natural language processing, computer vision, and tasks involving multimodal data. In these scenarios,

models need to generalize across multiple domains or modalities to address complex zero-shot learning challenges.

2.3 Image Generation

In the realm of image generation, various techniques and models have been explored to create realistic and diverse images. This section provides an overview of the related work in the field, highlighting key methodologies that have influenced and shaped the development of image generation models.

2.3.1 Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) [18] have been instrumental in the progress of image generation. Proposed by Ian Goodfellow and his colleagues in 2014, GANs consist of a generator and a discriminator engaged in a game-theoretic scenario. The generator aims to produce realistic images, while the discriminator strives to differentiate between real and generated images. This adversarial training process encourages the generator to continuously improve its image generation capabilities. GANs have demonstrated success in generating high-quality images across various domains.

2.3.2 Variational Autoencoders (VAEs)

Variational Autoencoders (VAEs) [19] provide a probabilistic approach to image generation. These models focus on learning the latent space representation of input images, capturing the underlying structure of the data. VAEs employ an encoder-decoder architecture, where the encoder maps input images to a probabilistic latent space, and the decoder reconstructs images from samples drawn from this space. VAEs are valued for their ability to generate diverse images by sampling from the learned latent space.

2.3.3 Generative Models for Image Synthesis

The high-dimensional nature of image data poses unique challenges for generative modeling. While Generative Adversarial Networks (GANs) [18] enable efficient sampling of high-resolution images with good perceptual quality, they struggle with optimization difficulties and capturing the complete data distribution. On the other hand, probability-based approaches emphasize robust density estimation, leading to more stable optimization. Variational Autoencoders (VAEs) [19] and flow-based models achieve an efficient synthesis of high-resolution images, yet their sample quality falls short of GANs.

Due to the inclusion of almost imperceptible high-frequency details in pixel-based image representations, maximum likelihood training demands an excessive capacity for modeling these details, resulting in prolonged training times. Recently, Diffusion Probabilistic Models (DPMs [20]) have demonstrated state-of-the-art results in both density estimation and sample quality. The generative power of these models arises when their underlying neural backbone is implemented as a UNet [21], aligning well with the natural biases of similar image data. However, evaluating and optimizing these models in pixel space suffers from slow inference speeds and extremely high training costs. While the former can be mitigated with advanced sampling strategies, training on high-resolution image data consistently demands computationally expensive gradients.

To address these drawbacks, we employ the denoising diffusion implicit model (DDIM) [2], which reduces the computational cost of training, accelerating inference without significant compromise in synthetic quality.

2.3.4 Conditional Image Generation

Conditional image generation involves incorporating additional information, such as class labels or textual descriptions, to guide the generation process. This approach allows for the generation of images based on specific conditions, adding a level of control and customization. Models like Conditional GANs (cGANs) and Conditional Variational Autoencoders (cVAEs)

have been employed for tasks requiring conditional image synthesis.

2.3.5 Conditional Diffusion Models

Conditional Diffusion Models [22–26] signify a noteworthy progression in the realm of image generation, synergizing the capabilities of diffusion processes with conditional generation methodologies. In contrast to conventional diffusion models, which predominantly emphasize unsupervised generation, conditional diffusion models introduce a nuanced layer of control and specificity through the incorporation of conditional information, such as class labels or textual descriptions, into the diffusion process.

The hallmark of conditional diffusion models lies in their proficiency to create images contingent upon specific conditions, thereby facilitating targeted and image synthesis. This methodology provides a pivotal instrument for applications necessitating meticulous command over the generated content. Incorporating conditioning information during the diffusion process enables the model to create images that match predefined traits, making it suitable for various tasks, including synthesizing defects in new components.

The formulation of conditional diffusion models incorporates the essence of the diffusion process, which can be expressed mathematically as follows:

$$P(X_t|X_{t-1}, ..., X_0, C) = \mathcal{N}(X_t; \mu(X_{t-1}, C), \sigma^2(X_{t-1}, C))$$

In this formula, X_t represents the image at time t, C denotes the conditioning information, μ signifies the mean function, and σ^2 denotes the variance function.

Conditional diffusion models show promise in addressing challenges related to defect generation, particularly in scenarios requiring precise control over defect features. Through the careful use of conditional information, these models not only enhance their capacity to generate images with realistic visual attributes but also dutifully adhere to specified conditions, thereby making a significant contribution to the borad field of advanced image generation techniques.

2.4 Language Model

Language models play a pivotal role in natural language processing and generation tasks. With the rapid advancement of deep learning, language models have seen significant improvements in performance and capabilities. In this section, we will explore language models related to defect generation, particularly those widely applied for generating descriptive text.

In recent years, state-of-the-art pre-trained language models such as GPT-3 (Generative Pre-trained Transformer 3) [27] and BERT (Bidirectional Encoder Representations from Transformers) [28] have achieved tremendous success in language understanding and generation tasks. These models, trained on massive datasets and equipped with powerful architectures, excel at comprehending and generating more natural, context-aware text.

In defect generation, we have witnessed remarkable progress with the integration of Contrastive Language-Image Pre-Training(CLIP) [3]. This innovative approach involves training language models not only on textual data but also on image data, fostering a more comprehensive understanding of the relationship between textual descriptions and corresponding visual content.

Especially in the context of new component defect generation, leveraging language models can aid in generating text descriptions that better match specific defect conditions. This ability is crucial for guiding defect generation models in producing more contextually meaningful images.

In contrast to traditional prompts that required detailed descriptions, our method does not require complex text but can still create accurate and exceptional images, even in new situations. This simplifies the generation process while ensuring both accuracy and diversity in the images produced.

2.5 Data Imbalance

In our research, we grapple with the challenge of data imbalance, where there is a significant disparity in the number of samples between normal and defective components. This imbalance

can potentially impact the training of the model, especially concerning the task of generating defects in new components.

Analysis of our dataset reveals a substantial difference in the quantity of normal and defective component samples, with a considerably larger number of normal samples. This scenario might lead the model to be overly biased towards learning features of normal components, resulting in suboptimal performance in generating defective components. To address this issue, we introduce oversampling techniques during data preprocessing, particularly focusing on oversampling defective component samples to achieve a balanced distribution between different categories.

Our oversampling methods involve replicating existing defective component samples to increase their representation in the dataset. This strategy aims to ensure that the model comprehensively learns the features of defective components, thereby enhancing the generation performance. It is crucial to note that excessive use of oversampling may pose the risk of overfitting, so we carefully tune the oversampling ratios during experiments.

Chapter 3

Method

In this chapter, a methodology employing the Compositional Conditional 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.

Section 3.1 involves the labeling of defect types and component group names, providing the model with embeddings formed by concatenating conditions (defect types) and component groups. The image features of component group X, compositional condition embeddings C, and time steps t are then input into the Unet [21] of the Diffusion model [29]. In Section 3.2, the overall architecture of the Compositional Conditional Diffusion Model(CCDM) is presented, demonstrating the fusion of compositional condition embeddings with the image features of defective components. Section 3.3 illustrates the forward process and diffusion process formulas of the CCDM. It also describes how Defect types and Component Groups are transformed from classes to embeddings. Time embedding is introduced into the Resblock [30] of U-Net, allowing time steps to be added to the image features. This aids U-Net in gradually denoising the Gaussian noise matrix during the diffusion process. Importantly, each predicted noise is guided by concatenated embeddings and time steps, directing the removal of random noise from the stochastic Gaussian noise matrix.

Finally, a series of experiments is conducted to verify the model's capability to generate defects for previously unseen components. Additionally, optimizations and improvements are applied to enhance the model's performance and applicability.

3.1 Labeling and Regrouping

In this study, we undertook the intricate process of data regrouping, aiming to classify components based on their similarities. The primary objective was to group components with subtle differences together, whilst recognizing instances where certain components exhibited exceptionally high degrees of similarity. Simultaneously, we became aware that achieving accuracy in fine-grained classification within this task is an independent and challenging area.

Considering these circumstances, we initially utilized a model pre-trained with MobileNetV3 [31] to automate the classification process. However, the outcomes generated by the model did not align with our expectations. Subsequently, we decided to incorporate human judgment in conjunction with the assistance of the model to label components. This hybrid approach, combining automated classification with human expertise, was adopted to enhance the precision and reliability of the labeling process, acknowledging the unique challenges posed by the intricate nature of the task.

3.1.1 Observer Consensus Process

To ensure consistent categorization/grouping, we invited multiple observers to conduct similarity assessments. These observers evaluated the similarity between components through shared guidelines and standardized interpretations. Such a consensus process helped eliminate subjective errors and increased the reliability of the grouping results.

3.1.2 Presentation of Grouping Results

The grouping results are presented in Figure 3.1, illustrating the restructured component groups. Each group represents the outcome of the observer consensus process, and components within each group are considered similar or related for the reader to comprehend the basis of these groupings.

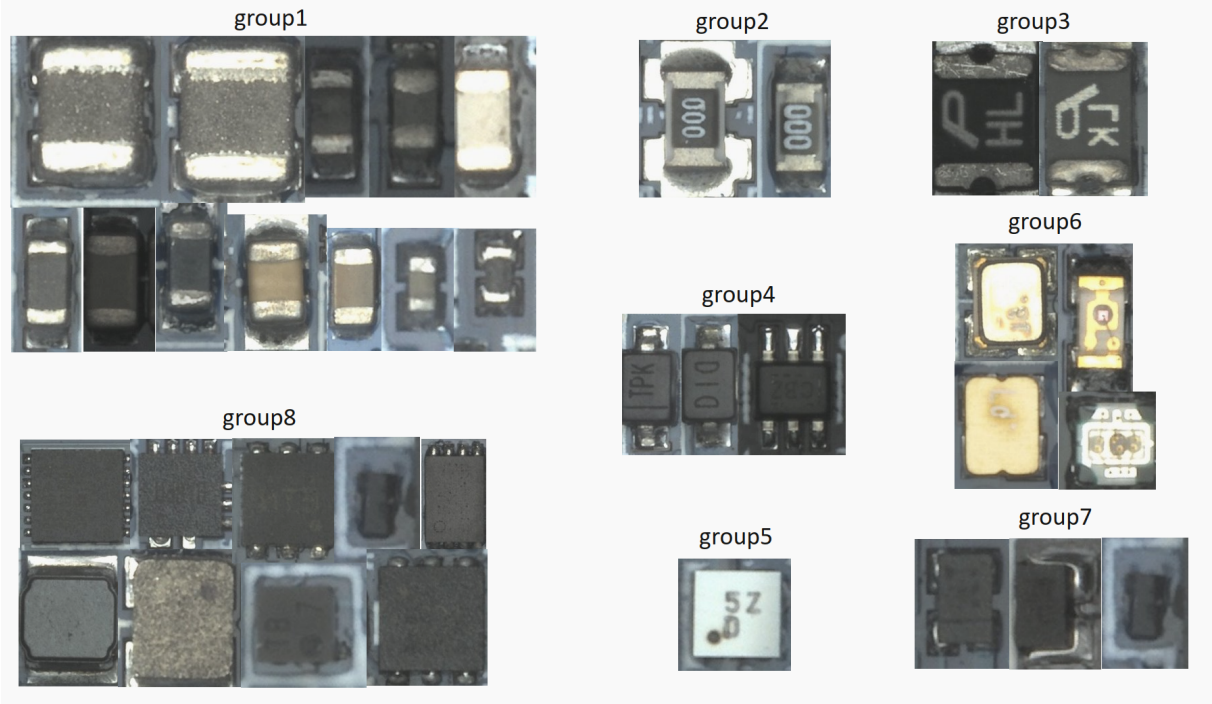

Figure 3.1: Presentation of Grouping Results

3.2 Model Architectures

In this context, we introduce our proposed CCDM in this chapter to generate images of previously unseen defects in new components, as illustrated in Figure 3-2.

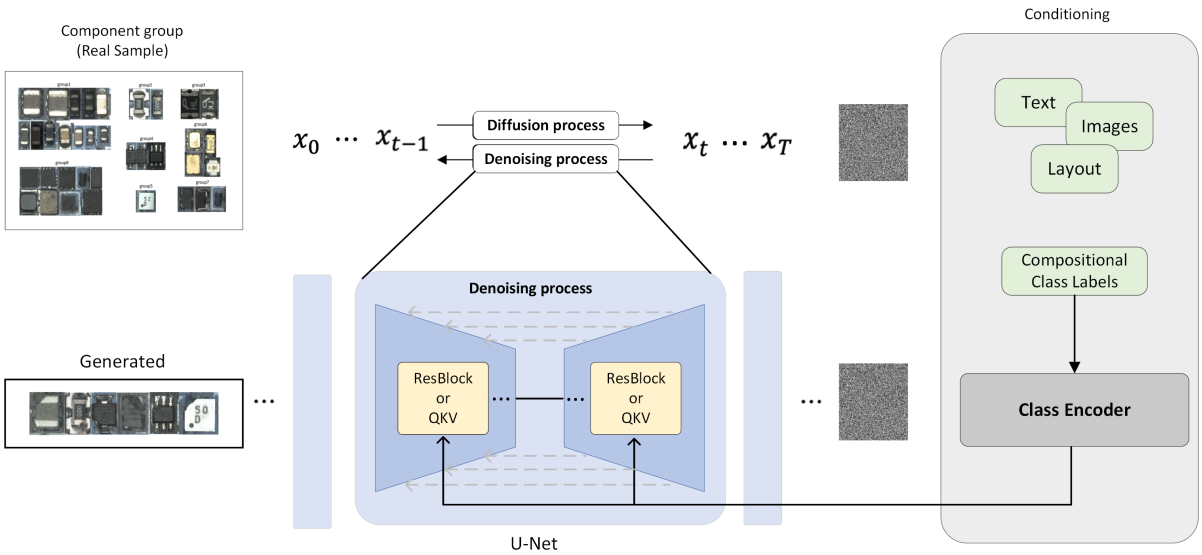

Figure 3.2: Compositional Conditional Diffusion Model(CCDM)

3.3 Conditional Diffusion Model

Denoising Diffusion Probabilistic Models (DDPM) [20] constitute a class of highly effective generative models in unconditional image generation. DDPM a learned Markov chain to transform a simple distribution, such as an isotropic Gaussian distribution, into the target data distribution. The generative process of DDPM involves learning the inverse process of the forward (diffusion) process a fixed Markov chain gradually adds noise to latent variables $x_1, ..., x_T$ sampled sequentially from the same dimensions.

In this context, let us consider having a sample x_0 from the distribution $D(x|c)$, where c serves as the compositional condition. The compositional condition c exhibits variability, and in this study, c is characterized by the amalgamation of attribute embedding and object embedding, denoted as $c \in concatenate(c_{obj}, c_{atr})$. This labeling approach signifies the integration of these two embeddings.Here, each step in the forward process is a Gaussian translation.

$$q(x_t|x_{t-1}, c) := \mathcal{N}(x_t \,|\, c; \sqrt{1-\beta_t}x_{t-1} \,|\, c, \beta_t I) \tag{3.1}$$

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.

$$q(x_t|x_0, c) := \mathcal{N}(x_t \,|\, c; \sqrt{\bar{\alpha}_t}x_0 \,|\, c, (1-\bar{\alpha}_t)I) \tag{3.2}$$

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

$$p_\theta(x_{t-1}|x_t, c) := \mathcal{N}(x_{t-1}\,;\, \mu_\theta(x_t, t, c); \Sigma_\theta(x_t, t, c)) \tag{3.3}$$

The variance μ_σ can be learnable parameters or a fixed set of scalars. As for the mean, after

reparameterization with $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$ for $\epsilon \sim \mathcal{N}(0, I)$, the loss function can be simplified as:

$$L := \mathbb{E}_{x_0,\epsilon}||\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t, c)|| \tag{3.4}$$

To incorporate the binary condition $c$into the U-Net architecture, we adopt a strategy inspired by [32]. This involves employing an embedding projection function, denoted as $e = f(c)$, where $f \in \mathbb{R} \rightarrow \mathbb{R}^n$, and n represents the embedding dimension. Subsequently, the condition embedding is added to feature maps across every Resblocks [22]. 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 [2]. 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.

$$x_{t-1}(x_t, t, c) = \sqrt{\bar{\alpha}_{t-1}}\frac{(x_t - \sqrt{1-\bar{\alpha}_t}\hat{\epsilon}(x_t, c)}{\sqrt{\bar{\alpha}_t}} + \sqrt{1-\bar{\alpha}_{t-1}}\hat{\epsilon_\theta}(x_t, c) \tag{3.5}$$

3.3.1 Conditioning Mechanisms

In principle, diffusion models model conditional distributions in the form of $D(x|c)$. To achieve this, we implement a conditional denoising autoencoder, denoted as $\epsilon_\theta(x_t, t, c)$, enabling control over the synthesis process through the input c. The widely recognized diffusion models the Stable Diffusion Model [1], 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, with grouping strategies outlined in Chapter 3.1.

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) [3] 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:

$$c_{obj} = Proj(Emb(Encoder(obj))) \tag{3.6}$$

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:

$$c_{atr} = Proj(Emb(Encoder(atr))) \tag{3.7}$$

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.

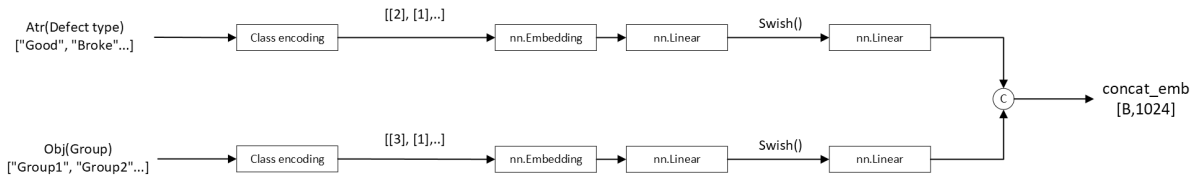

Figure 3.3: Condition Module

3.3.2 U-Net

We also experiment with a layer that we refer to as addition group normalization (AddGN), which incorporates the timestep and class embedding into each residual block after a group normalization [33] operation. We define this layer as $AddGN(h, y) = y_s + GroupNorm(h)$, 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.

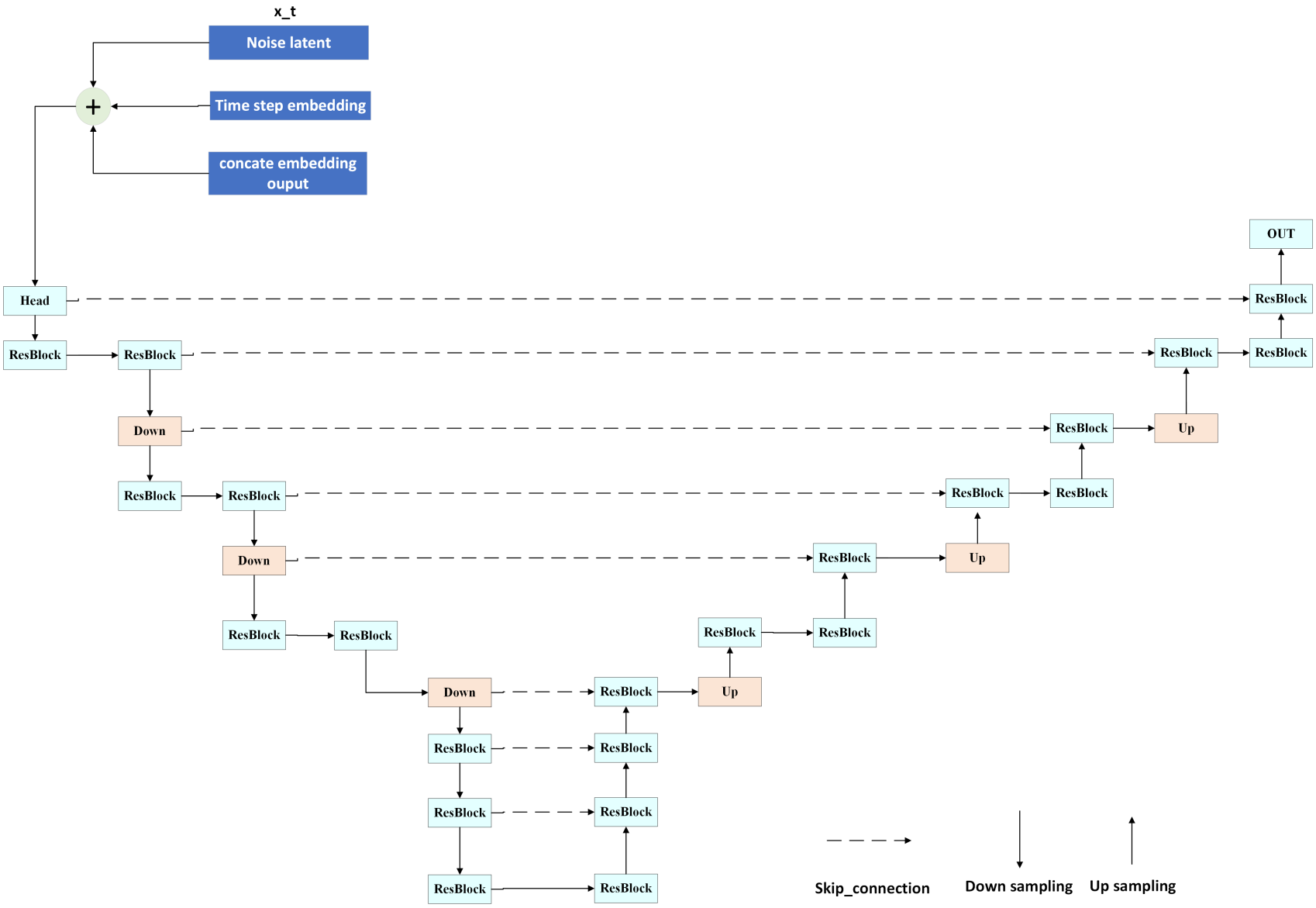

Figure 3.4: U-Net condition mechanisms(U-Net 1)

3.4 Algorithm

In the previous sections, we introduced the methods design for our Compositional Conditional Diffusion model,including Conditional Diffusion model, Conditioning Mechanisms, and U-Net condition mechanisms. In this section, we will present the combination of all these methods. The pseudocode for the proposed Compositional Class-to-image Diffusion model is shown in Algorithm 3.1 and 3.2.

Algorithm 3.1 Training a diffusion model with classifier-free guidance

Require: p_{uncond}: probability of unconditional training

1: **repeat**

2: $(x_0, c) \sim q(x_0, c)$

3: $c \leftarrow \emptyset$ with probability p_{uncond}

4: $\epsilon \sim \mathcal{N}(0, I)$

5: $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{(1 - \bar{\alpha}_t)}\epsilon$

6: Take gradient descent step on $\nabla_\theta ||\epsilon_\theta(x_t, c, t)) - \epsilon||^2$

7: **Until** converged

The algorithm denoted as 3.1 in this research pertains to the training procedure. It involves iteratively sampling from a standard Gaussian distribution to obtain a noise term, denoted as ϵ. This epsilon is then combined with an initial image x_0 through a forward process, resulting in a noisy image x_t. Subsequently, the generated noisy image x_t, along with a conditioning variable c, is fed into the model to predict the associated noise. The primary objective of the training process is to minimize the disparity between the predicted noise and the actual noise added during the generation process.

Algorithm 3.2 Sampling

Require: w: probability of unconditional training

1: $x_T \sim \mathcal{N}(0, I)$, $c \sim p(c)$

2: **for** t = T, …, 1 **do**

3: $z \sim \mathcal{N}(0, I)$if $t > 1$,else z = 0

4: $\tilde{\epsilon}_\theta = (w+1)\epsilon_\theta(x_t, c, t) - w\epsilon_\theta(x_t, t)$

5: $x_{x-1} = \sqrt{\alpha_{t-1}}(\frac{x_t - \sqrt{1-\alpha_t}\tilde{\epsilon}_\theta}{\sqrt{\alpha_t}}) + \sqrt{1 - \alpha_{t-1} - \sigma_t^2} \cdot \tilde{\epsilon}_\theta + \sigma_t \epsilon_t$

6: **end for**

7: **return** x_0

Algorithm 3.2 pertains to the sampling process. To obtain the image x_{t-1}, the reverse process outlined in Equation (3.5) is applied. This involves subtracting the noise predicted by the model from the current image x_t, multiplying by certain coefficients, and ultimately adding a noise term z, We then perform sampling using the following linear combination of the conditional and unconditional score estimates [26].

$$\epsilon_\theta = (w+1)\epsilon_\theta(x_t, c, t) - w\epsilon_\theta(x_t, t) \tag{3.8}$$

By following this procedure, the desired image x_{t-1} is obtained. This sampling algorithm is integral to the overall framework, enabling the generation of sequential images by iteratively applying the reverse process to generate each successive image in the sequence. The incorporation of model-predicted noise, along with carefully tuned coefficients, ensures the accuracy of the generated images and aligns with the overarching objective of achieving realistic and high-quality image synthesis [34].

3.5 Different U-Net Architectures

In this study, we further explored an alternative architectural design to investigate its performance. The detailed architecture is presented below:

Our inspiration originates from the control of conditions in Stable Diffusion. Initially, they employ a pre-trained Language-Image model known as Contrastive Language-Image Pre-Training (CLIP) [3]. This model learns the relationship between a complete sentence and the image it describes. In other words, during training, given an input sentence, the model becomes proficient in retrieving the most relevant images corresponding to that sentence.

Motivated by this model, we delve into the study of a model that establishes a connection between images and classes based solely on class information. Referred to as cMLIP (Compositional Multi Label-Image Pre-training) We demonstrate in Figure 3.5, this model is designed to link images and classes.

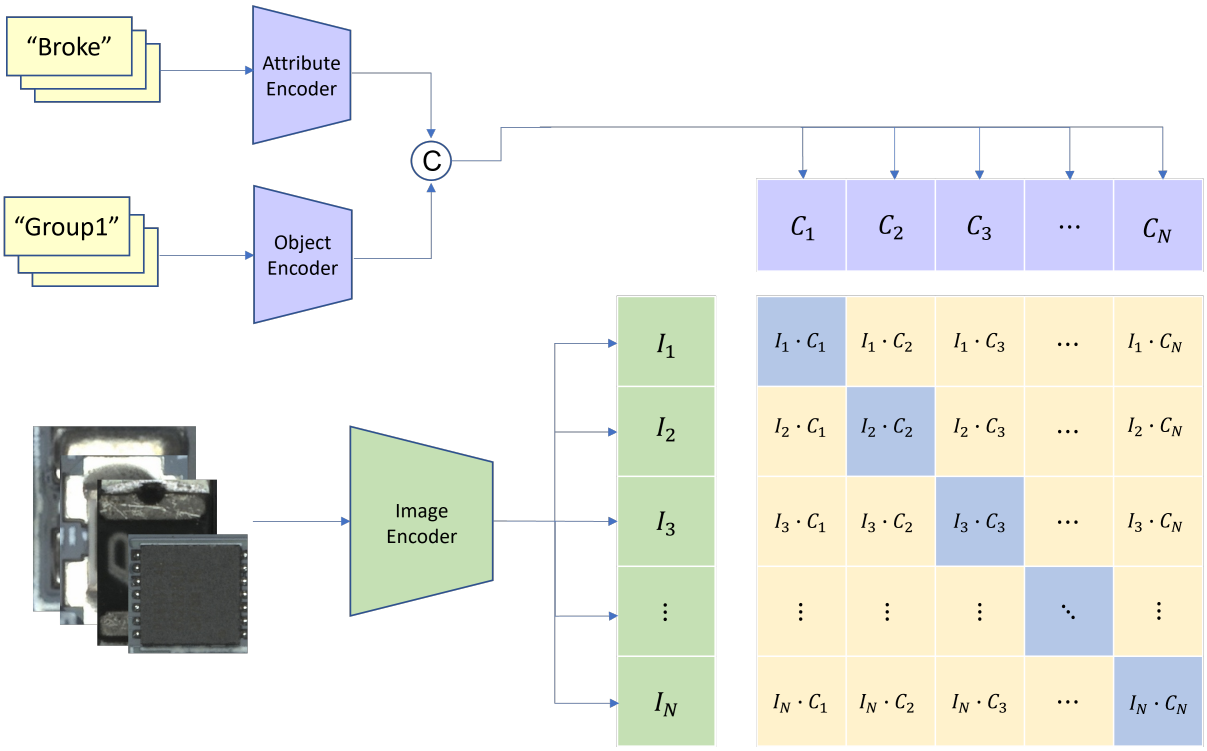

Figure 3.5: Contrastive Compositional Multi Label-Image Pre-training(cMLIP)

Subsequently, we utilize cMLIP in conjunction with a diffusion model to generate images corresponding to specific conditions. This innovative approach draws from the principles of Stable Diffusion and leverages the pre-trained cMLIP model to produce meaningful and self-attention and cross-attention mechanisms, facilitating the fusion of the concatenated embeddings with the Image feature of the defective component condition-controlled image synthesis. Figure

3.6 illustrates the mechanisms of the U-Net with cMLIP.

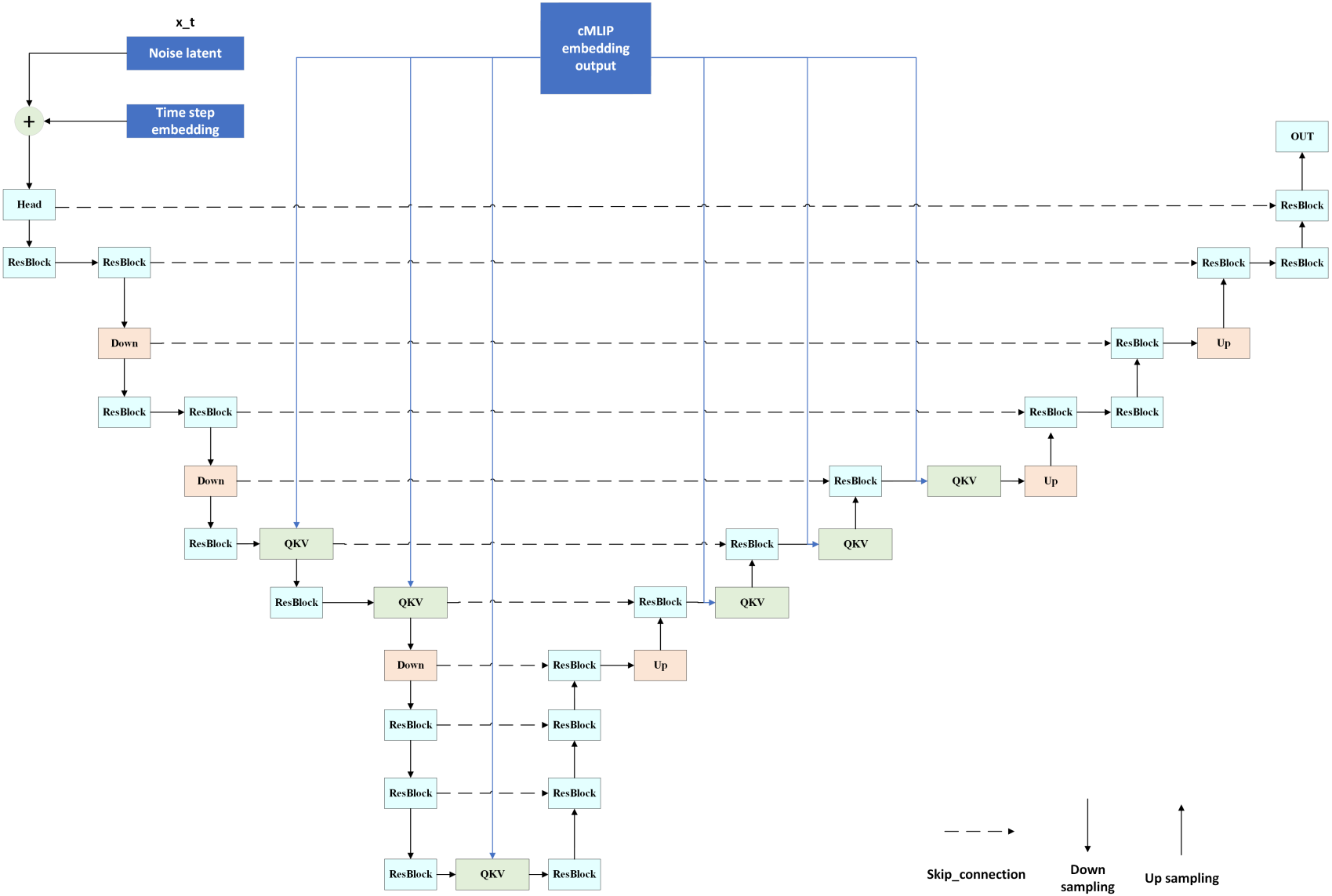

Figure 3.6: U-Net condition mechanisms(U-Net 2)

3.6 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 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.

Chapter 4

Experiments and Results

4.1 Dataset

This research utilizes a welding 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 welding 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 1.1.

It is important to note that the ”Good” category contains a substantial number of component images, creating a noticeable imbalance compared to the relatively limited 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. For a more comprehensive understanding of this reorganization process, please refer to Section 3.1.

We generated a simplified yet representative dataset using OpenCV for training purposes. Distinguishing features within this dataset include variations in color intensity and changes in shape, which are considered unique attributes for different groups. Furthermore, defect types across different groups are characterized by the absence of corners and rotations, ranging from angles greater than 1 degree to less than 45 degrees. Each combination comprises 500 images, resulting in a total of 3 * 5 * 500 images, as illustrated in Figure 4.1.

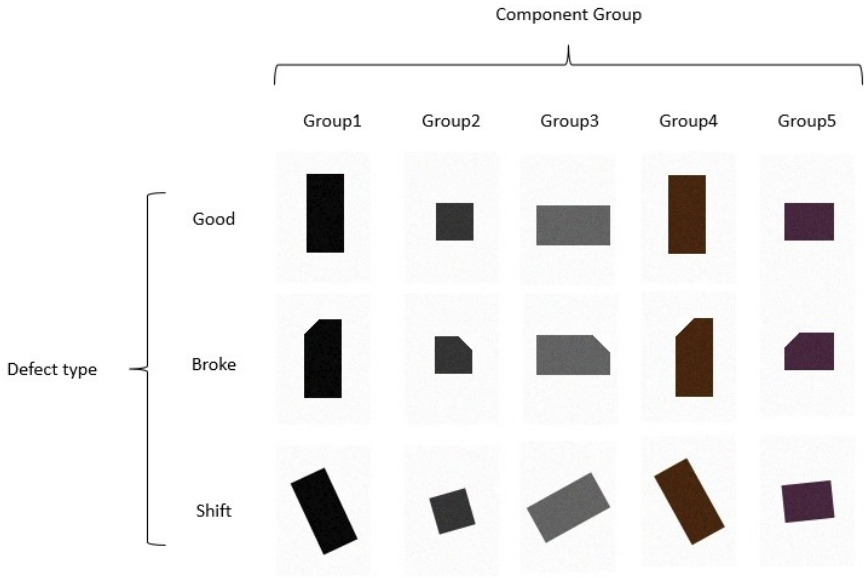

Figure 4.1: Toy Dataset

4.2 Experimental Setup

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/10.

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.

4.3 Experiment Results

The Compositional Conditional Diffusion Model (CCDM) provides a flexible and computationally tractable approach for synthesizing images across various modalities. We provide detailed information on the architecture, implementation, training, and evaluation of all the results presented in this section.Additionally, an overview of the hyperparameters of all trained CCDM is provided in this section.

4.3.1 Toy Dataset

Initially, we started by removing the instances from the Toy Dataset (as shown in Figure 4.1) corresponding to the ”Broke Group1” and ”Shift Group1.” This exclusion ensures that these groups do not contribute to the training add a space in between the period and subsequently, we fed these modified datasets into the cMLIP model for training, striving to develop a pre-trained class-image model for utilization as an embedder in the conditional diffusion model.Tab 4.1 illustrates the hyperparameters employed in training the cMLIP model.

cMLIP	
Epoch	10
Batch size	16
Learning rate	1e-4
Dropout	0.15
Weight Decay	1e-4
Class embedding dimension	512
Projection dimension	256

Table 4.1: The parameter settings of cMLIP in Toy Dataset

After training, we compute the similarity between cMLIP image features and class features, followed by applying softmax. As shown in Tab 4.2, the images labeled as ’Broke Group1’ and

'Shift Group1,' which are not present in the dataset, exhibit remarkably high probabilities in terms of accurate predictions.

cMLIP result		
Group name	Top1(%)	Top5(%)
Good Group1	100	100
Good Group2	100	100
Good Group3	100	100
Good Group4	100	100
Good Group5	100	100
Broke Group1 (unseen)	97	100
Broke Group2	100	100
Broke Group3	100	100
Broke Group4	100	100
Broke Group5	100	100
Shift Group1 (unseen)	76	100
Shift Group2	100	100
Shift Group3	100	100
Shift Group4	100	100
Shift Group5	100	100

Table 4.2: The outcomes of cMLIP in the Toy Dataset

Compositional Conditional 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 4.3: The parameter settings of CCDM in Toy Dataset

In Figure 4.2, we can observe that both the U-Net 1 and U-Net 2 methods exhibit commendable performance in generating unseen images on the toy dataset. Through validation on this toy dataset, we establish the feasibility of these two approaches.

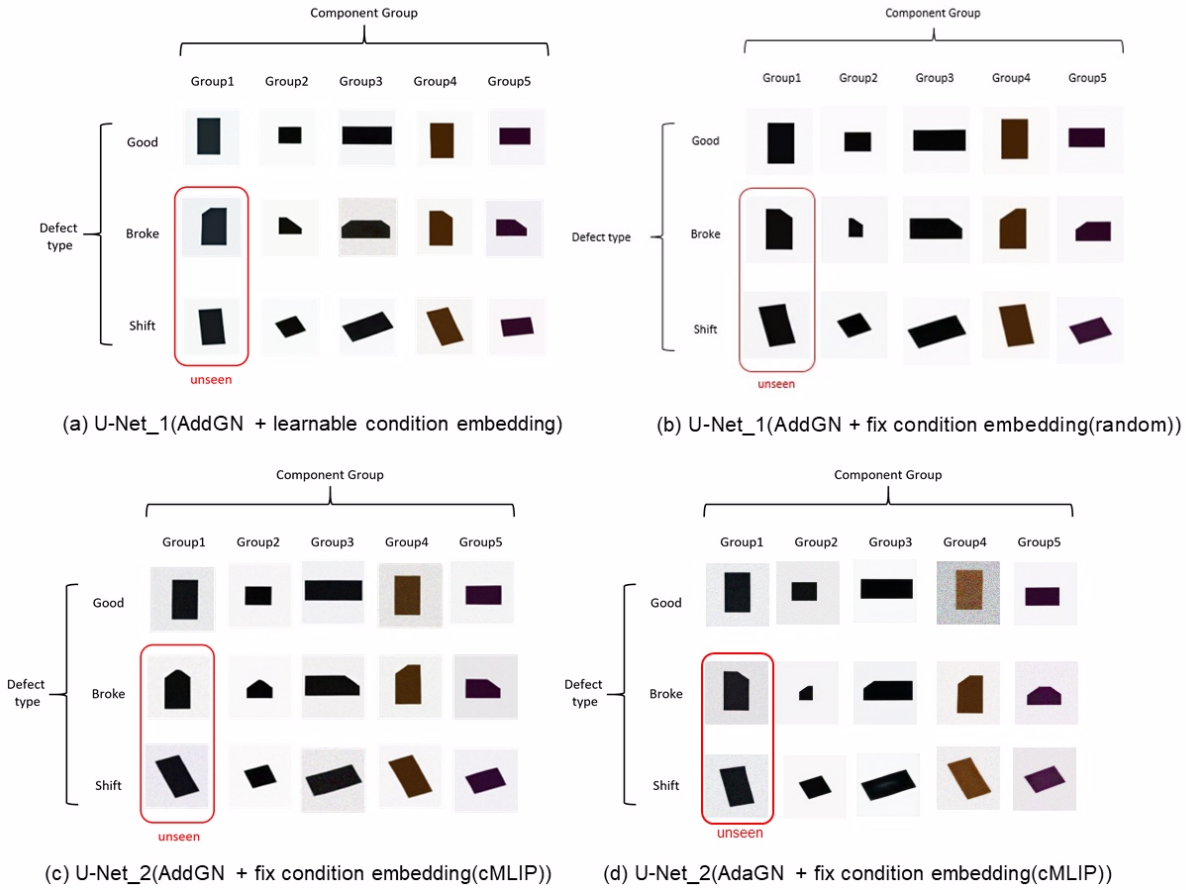

Figure 4.2: On the Toy dataset, the results generated by CCDM are presented using (a) the U-Net 1 method with learnable condition embedding, (b) the U-Net 1 method with fixed condition embedding(random), (c) the U-Net 2 method with fixed condition embedding (cMLIP), and (c) the U-Net 2 (AdaGN) method with fixed condition embedding (cMLIP).

Upon careful examination of the outcomes, it is evident that using the U-Net 1 method with learnable condition embedding demonstrates superior performance in generating unseen images on the Toy dataset.

4.3.2 PCB Dataset

As depicted in Figure 1.1, however, we observed distinct patterns for the Broke samples in each Group. We specifically focused on the Broke pattern associated with missing corners. Therefore, we selected several Groups, namely Group1, Group4, Group5, and Group7, where the Broke patterns aligned with our criteria. The subsequent presentation aims to showcase the results obtained on the PCB dataset. Identical hyperparameters as the toy dataset were utilized

during the training of cMLIP. Let us now delve into the details presented in Table 4.4.

cMLIP	
Epoch	10
Batch size	16
Learning rate	1e-4
Dropout	0.15
Weight Decay	1e-4
Class embedding dimension	512
Projection dimension	256

Table 4.4: The parameter settings of cMLIP in PCB Dataset

Due to the absence of Broke Group3 in the PCB Dataset, the calculation of its cosine similarity in cMLIP is not feasible. Nevertheless, it is evident that cMLIP demonstrates impressive performance on other observed data points.

cMLIP result		
Group name	Top1(%)	Top5(%)
Good Group1	100	100
Good Group3	97	100
Good Group4	100	100
Good Group5	100	100
Good Group7	100	100
Broke Group1	96	100
Broke Group3 (unseen)		
Broke Group4	100	100
Broke Group5	100	100
Broke Group7	100	100
Shift Group1	100	100
Shift Group3	100	100
Shift Group4	100	100
Shift Group5	97	100
Shift Group7	99	100

Table 4.5: The outcomes of cMLIP in the PCB Dataset

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 4.6: The parameter settings of CCDM in PCB Dataset

In Figure 4.3, we can observe that both the U-Net 1 and U-Net 2 methods exhibit commendable performance in generating unseen images on the PCB dataset.

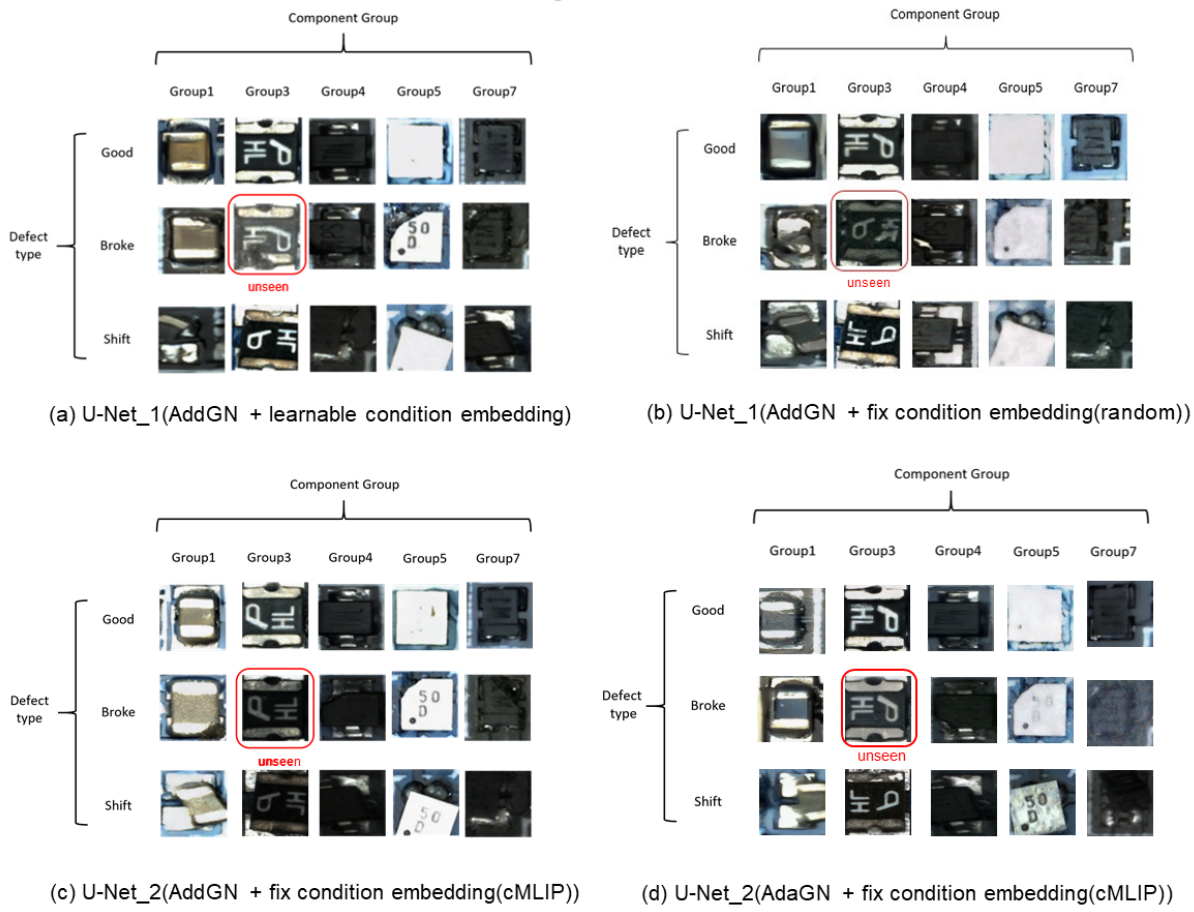

Figure 4.3: On the PCB dataset, the results generated by CCDM are presented using (a) the U-Net 1 method with learnable condition embedding, (b) the U-Net 1 method with fixed condition embedding(random), (c) the U-Net 2 method with fixed condition embedding (cMLIP), and (c) the U-Net 2 (AdaGN) method with fixed condition embedding (cMLIP)

During the PCB dataset image sampling process, we observed that U-Net 2 and U-Net 1 with fix embedding are failed to generate images accurately, whereas U-Net 1 with learnable condition embedding could generate images correctly, albeit with a high error rate. Some generated images exhibited discrepancies with the specified compositional conditions. In Figure 4.4, we sampled 50 images of unseen images (broke group 3), revealing a relatively low accuracy. 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.

Figure 4.4: The highlighted portion in the red box represents the results selected by the binary classifier, further refined through manual curation.

Due to the lack of actual Broke Group3 samples, we conducted an experiment using ResNet152 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 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.

Finally, I added some raw images from Broke Group1 into the training data so that the dataset would not be entirely without Broke Group1, mimicking the typical structure of real data. Initially, the result was 97%. After incorporating photos generated by our CCDM as an augmented dataset, the accuracy improved to 98%.

This result demonstrates our success in generating new component defect images, which serves as an expansion of the dataset for defect detection applications.Figure 4.5 is generated by selecting an image produced by U-Net 1 with learnable condition embedding, specifically from broke Group3.

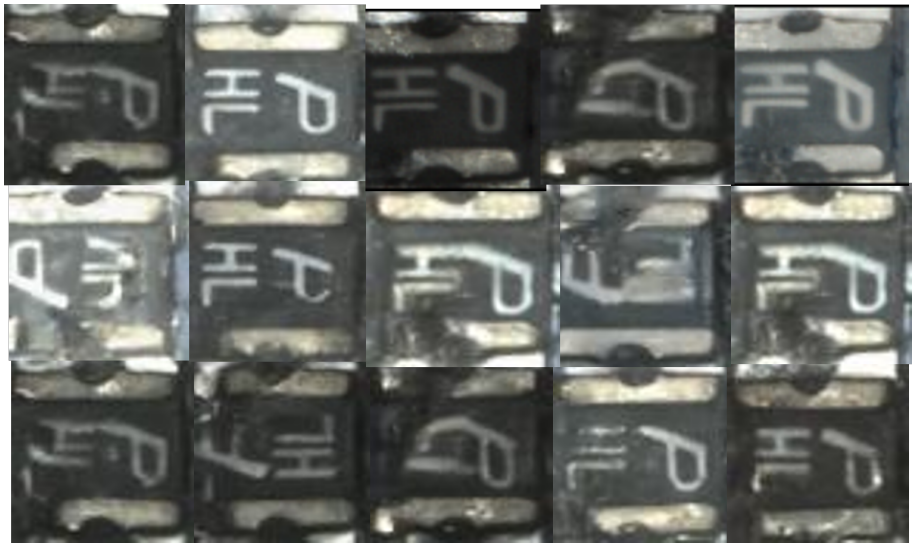

Figure 4.5: The image generated after selecting an image using U-Net 1 with learnable condition embedding.

Chapter 5

Conclusion

In this paper, we proposed an innovative concept of a compositional conditional diffusion model to generate unseen defect component. We in a method based on class-image correlation to regulate the diffusion model for image generation. The outcomes of this research can be applied in the context of defect detection in electronic industry production lines. When encountering new defective components in the future, the compositional conditional diffusion model can be employed to generate visual representations of these components. The produced images can then be incorporated into the defect detection model for further training.

This research project aims to investigate the practical aspects of human composite cognitive abilities, emphasizing the process of learning from existing composite concepts and applying them to novel composite concepts. Specifically, in the domain of image generation, the goal is to achieve ”composite zero-shot image generation and selection.” Within the academic realm of artificial intelligence, the study delves into the generalization capabilities of neural networks in the context of composite zero-shot learning and generation. We look forward to further exploring the compositional condition diffusion model in a wider variety of settings and data modalities.

References

[1] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," pp. 10 684–10 695, 2022.

[2] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[3] 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," pp. 8748–8763, 2021.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[5] M. Mancini, M. F. Naeem, Y. Xian, and Z. Akata, "Open world compositional zero-shot learning," pp. 5222–5230, 2021.

[6] G. Xu, P. Kordjamshidi, and J. Y. Chai, "Zero-shot compositional concept learning," *arXiv preprint arXiv:2107.05176*, 2021.

[7] A. Panda and D. P. Mukherjee, "Compositional zero-shot learning using multi-branch graph convolution and cross-layer knowledge sharing," *Pattern Recognition*, vol. 145, p. 109916, 2024.

[8] M. Yang, C. Xu, A. Wu, and C. Deng, "A decomposable causal view of compositional zero-shot learning," *IEEE Transactions on Multimedia*, 2022.

[9] X. Lu, S. Guo, Z. Liu, and J. Guo, "Decomposed soft prompt guided fusion enhancing for compositional zero-shot learning," pp. 23 560–23 569, 2023.

[10] N. Saini, K. Pham, and A. Shrivastava, "Disentangling visual embeddings for attributes and objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 13 658–13 667.

[11] W. Wang, V. W. Zheng, H. Yu, and C. Miao, "A survey of zero-shot learning: Settings, methods, and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–37, 2019.

[12] F. Pourpanah, M. Abdar, Y. Luo, X. Zhou, R. Wang, C. P. Lim, X.-Z. Wang, and Q. J. Wu, "A review of generalized zero-shot learning methods," *IEEE transactions on pattern analysis and machine intelligence*, 2022.

[13] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 9, pp. 2251–2265, 2018.

[14] S. Rahman, S. Khan, and F. Porikli, "A unified approach for conventional zero-shot, generalized zero-shot, and few-shot learning," *IEEE Transactions on Image Processing*, vol. 27, no. 11, pp. 5652–5667, 2018.

[15] W.-L. Chao, S. Changpinyo, B. Gong, and F. Sha, "An empirical study and analysis of generalized zero-shot learning for object recognition in the wild," pp. 52–68, 2016.

[16] Y. Xian, T. Lorenz, B. Schiele, and Z. Akata, "Feature generating networks for zero-shot learning," pp. 5542–5551, 2018.

[17] Z. Han, Z. Fu, S. Chen, and J. Yang, "Contrastive embedding for generalized zero-shot learning," pp. 2371–2381, 2021.

[18] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[19] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” *arXiv preprint arXiv:1312.6114*, 2013.

[20] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[21] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” pp. 234–241, 2015.

[22] P. Dhariwal and A. Nichol, “Diffusion models beat gans on image synthesis,” *Advances in neural information processing systems*, vol. 34, pp. 8780–8794, 2021.

[23] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, “Score-based generative modeling through stochastic differential equations,” *arXiv preprint arXiv:2011.13456*, 2020.

[24] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, W. Zhang, B. Cui, and M.-H. Yang, “Diffusion models: A comprehensive survey of methods and applications,” *ACM Computing Surveys*, vol. 56, no. 4, pp. 1–39, 2023.

[25] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, “Diffusion models in vision: A survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[26] J. Ho and T. Salimans, “Classifier-free diffusion guidance,” *arXiv preprint arXiv:2207.12598*, 2022.

[27] N. Dehouche, “Plagiarism in the age of massive generative pre-trained transformers (gpt-3),” *Ethics in Science and Environmental Politics*, vol. 21, pp. 17–23, 2021.

[28] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.

[29] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, “Deep unsupervised learning using nonequilibrium thermodynamics,” pp. 2256–2265, 2015.

[30] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” pp. 770–778, 2016.

[31] B. Koonce and B. Koonce, “Mobilenetv3,” *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*, pp. 125–144, 2021.

[32] P. Dhariwal and A. Nichol, “Diffusion models beat gans on image synthesis,” *Advances in neural information processing systems*, vol. 34, pp. 8780–8794, 2021.

[33] Y. Wu and K. He, “Group normalization,” pp. 3–19, 2018.

[34] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, “Score-based generative modeling through stochastic differential equations,” *arXiv preprint arXiv:2011.13456*, 2020.