参赛队员姓名：王郡阳、刘忠恕

中学：广东华南师范大学附属中学

省份：广东省

国家/地区：中国

指导教师姓名：谭平、杨晓安

指导教师单位：香港科技大学（HKUST）电子与计算机工程系、广东华南师范大学附属中学

论文题目：Trans3-Vision: Transfer Learning based Transformer for Transparent Object Segmentation with Grounded-ID2

Trans3-Vision: Transfer Learning based Transformer for Transparent Object Segmentation with Grounded-ID2

Junyang Wang, Zhongshu Liu
Affiliated High School of South China Normal University
Email: {wjy.michael, liuzs.james}2022@gdhfi.com

Abstract

Autonomous driving presents a captivating future application of AI. With the increasing accessibility of AI software and hardware, we embarked on assembling an autonomous driving vehicle with the help of online tutorials and open-source resources. Our objective was to explore the intriguing applications of autonomous driving technology within the campus environment.

However, we discovered a limitation in our autonomous vehicle's online mapping ability to recognize glass walls, a material extensively used as classroom partitions on our campus. Investigation revealed that laser beams of the vehicle's LiDAR system pass through transparent glass without effective reflection, resulting in collisions with glass walls under certain conditions. As much modern architecture incorporates transparent glass materials, this issue hinders the application of autonomous driving technology around these new environments. Therefore, in this research, we developed a training method for a machine vision AI model to detect transparent objects.

During the research process, we primarily addressed the following challenges:

- We developed an **automatic mask-labeling data generation methodology and pipeline (Integrated Pipeline of Stable Diffusion Inpainting with Grounded-SAM)**, which solves the lack of image data needed for semantic segmentation tasks from any research area, alleviates human manual work and lowers overall time consumption on creating datasets. With our methodology, we effortlessly created **Grounded-SAM Integrated Diffusion Inpainting Dataset (Grounded-ID2)**, a real-world-simulating glass segmentation dataset.
- Proposed an improved Transfer learning-based Transformer for the Transparent Object Segmentation model (Tran3-Vision) to overcome the performance losses trained with combined datasets. Trans3-Vision performs better on the test data set from Trans10kv2 with a mIoU score of 75.94, **a 2.1% increase over other state-of-the-art models**. Trans3-Vision utilized domain adaption, an optimized training process of transfer learning, in association with Grounded-ID2.

The Grounded-ID2 dataset proved a vital asset in improving the performance of Trans3-Vision. The process introduced by Grounded-ID2 is not limited to glass and transparent object segmentation **and applies to all other areas where image masks are needed**.

This project is made open-source on: https://github.com/PROMCRdog/Trans3-Vision and the Current Grounded-ID2 Glass dataset is uploaded and will be updated on : Dropbox.

Keywords: *Object Segmentation, Automatic Labeled Data Generation, Domain Adaption, Stable Diffusion, Segment Anything, Image Inpainting, Vision Transformer*

Trans3-Vision：利用 Grounded-ID2 并基于迁移学习和 Transformer 的透明物体分割识别

王郡阳，刘忠恕
广东华南师范大学附属中学
{wjy.michael, liuzs.james}2022@gdhfi.com

摘要

自动驾驶技术是人工智能领域最前沿的一项应用。随着开源软件和硬件的日益普及，我们利用在线教程和开源资源组装了一辆自动驾驶小车。我们最初的目标是看看自动驾驶小车在校园环境中有哪些有趣的应用。

然而，小车组装好后，我们发现自动驾驶小车在我们的校园中存在严重问题，即它无法识别校园中无处不在的透明玻璃墙，导致小车经常撞向玻璃。经过研究我们发现，问题的原因是车辆的激光雷达系统发出的激光会穿过玻璃，无法形成有效的反射；而相机系统和识别算法也同样无法识别透明的玻璃。如今，在学校、商场、办公楼等很多现代建筑室内设计中玻璃都被大量使用。这个问题阻碍了自动驾驶技术在这些室内环境中的应用。经过调研，我们发现业界目前还没有针对透明玻璃墙的成熟自动驾驶解决方案。因此，在此研究中，我们开发了一套能让基于机器视觉的人工智能模型检测透明物体的算法。

在研究过程中，我们主要解决了以下挑战：

- 我们开发了一套 AI 自动生成标注数据集的方法和流程范式（**Integrated Pipeline of Stable Diffusion Inpainting with Grounded-SAM**），从而解决了各类图像分割领域中数据缺乏的问题，使得我们能快速创建大规模的数据集，用于有效的训练和应用，节省了人工创建数据集所耗费的时间与精力。我们借助该技术创建了一套供玻璃分割用的仿真数据集 **Grounded Integrated Diffusion Inpainting Dataset**（简称为 **Grounded-ID2**）。
- 我们提出了一种改进的，基于迁移学习的透明物体分割 Transformer 模型（Trans3-Vision），以解决使用合成数据集与现实世界存在差异导致性能下降的问题。Trans3-Vision 在来自 Trans10kv2 的测试数据集上表现更好，mIoU 得分为 75.94，**比其他最先进的模型提高了 2.1%**。Trans3-Vision 结合了 Domain Adaption 技术和优化的迁移学习训练过程，使得它也能够很好的用于我们的 Grounded-ID2 数据集。

Grounded-ID2 数据集在提高 Trans3-Vision 的性能方面发挥了重要作用。因此，Grounded-ID2 代表的 AI 合成训练数据所使用的流程范式不仅适用于透明物体分割和识别，还同样**适用于所有需要训练数据的图像分割领域**。

该项目代码发布在 https://github.com/PROMCRdog/Trans3-Vision，Grounded-ID2 数据集上传到 Dropbox 云盘。

关键词：物体分割识别、自动标注数据生成、深度域自适应、Stable Diffusion、Segment Anything、图像补全、ViT

Contents

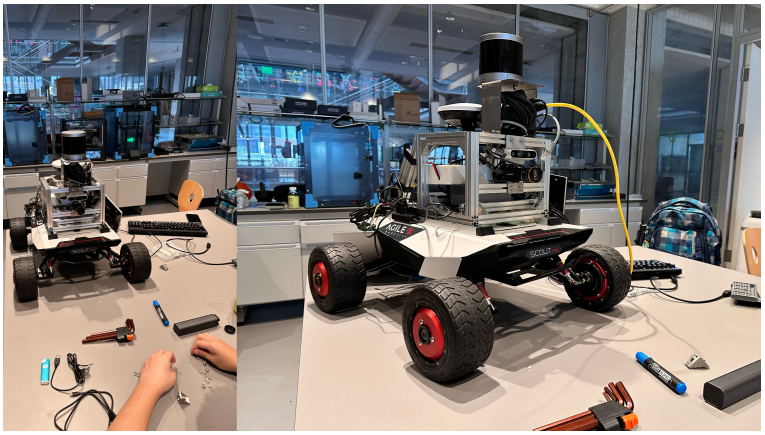


Figure 1: *The autonomous driving robotic car we built*

1 Introduction

1.1 Background

We spent months researching the necessary equipment for our robotic car, shown in Fig.1. We decided to use state-of-the-art hardware to explore at the forefront level. We wanted our car to achieve modern-day autonomous driving like Teslas and other tech start-ups. After designing our car structure and adapting our equipment into a single shell, we began our software development based on the open-source ROS (Robot Operating System). Many resources are credited to the open-source community surrounding the ROS development platform. Developed on the agilex robotics scout hardware platform, the car is equipped with an Nvidia Jetson AGX Orin as the primary computing power. It uses a Robosense Lidar sensor and a Hikonvision polarized camera. It is equipped with a GNSS locator, which is a precise timer for the whole system. **We used gmapping with our lidar sensors to generate an offline map in real-time. From Fig. 2, we quickly realized that glass walls and panels are a huge problem for our robotic car's navigation.** In the background of Fig.1, it is also evident how our school uses many glass wall partitions, which poses this challenging task we aim to address in our paper.

To validate our observations, commercial self-driving cars are also unable to detect glass. We drove a recently released Xpeng(小鹏) G6 equipped with XNGP—the best autonomous driving system in China—up to a glass door. As shown in Fig.3, there is no sign of any glass detection. The distance is

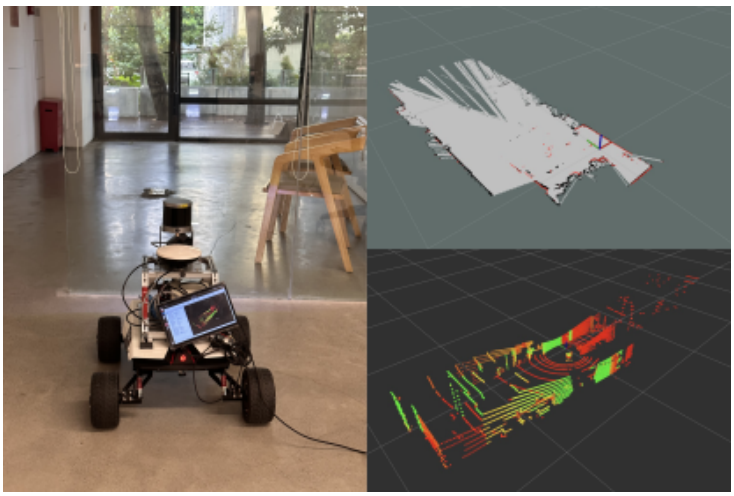


Figure 2: *The car's sensor outputs next to glass: gmapping (top right), lidar 3D-point-map (bottom right)*

only measured by the equipped low-resolution ultrasonic sensors of the wall next to the glass door. Even the best of the best autonomous driving systems can not detect glass.

In many domains, such as robotics, autonomous vehicles, and computer vision, the detection and segmentation of transparent and semitransparent materials present significant challenges for scene understanding and object recognition. Specifically, autonomous vehicles from Roombas to Teslas all fail to detect and model transparent surfaces. Transparent objects, such as glass surfaces, doors, and walls, are omnipresent in our daily lives, serving practical and decorative purposes. However, their existence poses critical problems for vision systems, impacting tasks such as depth prediction, instance segmentation, robotic navigation, and drone tracking. Failure to detect and accurately segment glass surfaces can lead to disastrous consequences, such as collisions and accidents. Unlike their opaque counterparts, transparent materials lack fixed patterns and exhibit complex dynamic appearances influenced by light-matter interactions, object shapes, and background factors. The interactions of light waves with fine materials give rise to reflection, refraction, and transmission effects, resulting in observations that are difficult to model and fall outside the distribution of typical data. This poses a fundamental challenge for existing scene understanding methods that heavily rely on texture-based cues. Current approaches for transparent material detection and segmentation primarily leverage contextual information or boundary detection in the RGB domain. However, the limited strength of cues in the RGB domain, caused by weak light-material interactions, makes accurate segmentation of transparent objects problematic. Although some research has explored richer representations of light-matter interactions, such as polarization, which relies on expensive specialized sensors. These methods are often hard to maintain and add additional work, limiting their practical applicability.

To address the detection and segmentation challenges posed by transparent materials, it is essential to develop novel approaches considering these materials' unique properties and behaviors. Leveraging both low-level cues (e.g., color differences and reflection artifacts) and high-level contextual cues (e.g.,

Figure 3: *Xpeng(小鹏) G6 Glass Door Visualization*

object relationships) can enhance the accuracy and robustness of vision systems in detecting and segmenting glass surfaces and other transparent objects. Many datasets were explicitly created to solve this problem, such as the GDD[1] and the Trans10k[2]. These datasets played a significant role in training past methods of glass detection; however, multiple limitations still exist within these datasets, allowing room for improvement in this aspect. For example, the Trans10k dataset consists of an imbalanced class distribution among the glass objects, which may lead to the under-representation of particular objects.

This paper aims to provide an improved solution to the transparent object detection and segmentation problem. First, we construct a novel Grounded-ID2 dataset that contains numerous AI-generated images of glass surfaces from diverse scenes for glass segmentation purposes. By utilizing the effectiveness of such AIGC (Artificial Intelligence Generated Content), we can create an infinite number of images with accurate labels and masks to boost the performance of our proposed model.

In addition, we proposed an enhanced model, Trans3Vision, to complete the glass detection and segmentation task. We integrate domain adaption methods into our segmentation model to boost the performance on unfamiliar datasets while maintaining accuracy on familiar ones.

Our main contributions can be summarized as follows:

- We have created and published **an AI-generated dataset specifically tailored for training autonomous driving models in the presence of transparent walls**. The approach, Grounded-ID2, enables the generation of an unlimited number of specialized datasets, enriching the driving experience of autonomous driving algorithms.
- An improved model **Trans3-Vision, utilizing DANN (Unsupervised Domain Adaptation by Backpropagation), is employed to achieve domain adaptation between different datasets**. Recognizing the disparities between AI-generated and real-world, we propose the Trans3Vision framework to enhance the adaptability and performance of our autonomous driving algorithms in real-world environments.

The rest of the paper is organized as follows: Section 2 reviews the related works. Section 3 introduces the AI-generated dataset for transparent wall detection. Section 4 explains the architecture of our Trans3-Vision algorithm. Section 5 compares the experiment results. Finally, Section 6 concludes the paper.

2 Related Works

In this section, we review the current state-of-the-art methods of transparent object/surface detection, classifying them based on the type of image used, including RGB image, RGB-Depth image, RGB-Thermal image, and RGB-Polarization image.

2.1 Methods using RGB

Methods utilizing only an RGB image to detect transparent objects often rely on specific contextual cues. These contextual cues may be classified into low-level cues (e.g., changes in the color hinting boundaries, blurred images/specular highlights caused by reflection) and high-level cues (correlations between many objects). Many successful methods take advantage of these cues to enhance the accuracy of their models. For example, GDNet, the first computational method for glass detection proposed by Mei et al.[1], uses a large-field contextual feature integration module to capture various contexts. Lin et

Table 1: *Comparing Trans4Trans to other State-of-the-art models*

Method	GFLOPs↓	ACC↑	mIoU↑	Category IoU↑											
				Background	Shelf	Jar/Tank	Freezer	Window	Door	Eyeglass	Cup	Wall	Bowl	Bottle	Box
FPENet[6]	0.76	70.31	10.14	74.97	0.01	0.00	0.02	2.11	2.83	0.00	16.84	24.81	0.00	0.04	0.00
ESPNetv2[7]	0.83	73.03	12.27	78.98	0.00	0.00	0.00	0.00	6.17	0.00	30.65	37.03	0.00	0.00	0.00
ContextNet[8]	0.87	86.75	46.69	89.86	23.22	34.88	32.34	44.24	42.25	50.36	65.23	60.00	43.88	53.81	20.17
FastSCNN[9]	1.01	88.05	51.93	90.64	32.76	41.12	47.28	47.47	44.64	48.99	67.88	63.80	55.08	58.86	24.65
DFANet[10]	1.02	85.15	42.54	88.49	26.65	27.84	28.94	46.27	39.47	33.06	58.87	59.45	43.22	44.87	13.37
Enet[11]	2.09	71.67	8.50	79.74	0.00	0.00	0.00	0.00	0.00	0.00	0.00	22.25	0.00	0.00	0.00
DeepLabv3+MBv2[12]	2.62	88.39	54.16	89.95	31.79	48.29	46.18	41.39	43.42	61.97	69.48	61.65	54.89	63.47	37.36
HRNet_w18[13]	4.20	89.58	54.25	92.47	27.66	45.08	40.53	45.66	45.00	68.05	73.24	64.86	52.85	62.52	33.02
HarDNet[14]	4.42	90.19	56.19	92.87	34.62	47.50	42.40	49.78	49.19	62.33	72.93	68.32	58.14	65.33	30.90
DABNet[15]	5.18	77.43	15.27	81.19	0.00	0.09	0.00	4.10	10.49	0.00	36.18	42.83	0.00	8.30	0.00
LEDNet[16]	6.23	86.07	46.40	88.59	28.13	36.72	32.45	43.77	38.55	41.51	64.19	60.05	42.40	53.12	27.29
Trans4Trans-T[2]	10.45	93.23	68.63	94.44	48.39	61.89	61.86	61.14	54.83	73.60	83.03	75.20	74.69	75.26	59.19
ICNet[17]	10.64	78.23	23.39	83.29	2.96	4.91	9.33	19.24	15.35	24.11	44.54	41.49	7.58	27.47	3.80
BiSeNet[18]	19.91	89.13	58.40	90.12	39.54	53.71	50.90	46.95	44.68	64.32	72.86	63.57	61.38	67.88	44.85
Trans4Trans-S[2]	19.92	94.57	74.15	95.60	**57.05**	71.18	**70.21**	63.95	61.25	81.67	87.34	78.52	**77.13**	81.00	64.88
DenseASPP[19]	36.20	90.86	63.01	91.39	42.41	60.93	64.75	48.97	51.40	65.72	75.64	67.93	67.03	70.26	49.64
DeepLabv3+[20]	37.98	92.75	68.87	93.82	51.29	64.65	65.71	55.26	57.19	77.06	81.89	72.64	70.81	77.44	58.63
FCN[21]	42.23	91.65	62.75	93.62	38.84	56.05	58.76	46.91	50.74	82.56	78.71	68.78	57.87	73.66	46.54
OCNet[22]	43.31	92.03	66.31	93.12	41.47	63.54	60.05	54.10	51.01	79.57	81.95	69.40	68.44	78.41	54.65
RefineNet[23]	44.56	87.99	58.18	90.63	30.62	53.17	55.95	42.72	46.59	70.85	76.01	62.91	57.05	70.34	41.32
Trans2Seg[4]	49.03	94.14	72.15	95.35	53.43	67.82	64.20	59.64	60.56	**88.52**	86.67	75.99	73.98	82.43	57.17
TransLab[24]	61.31	92.67	69.00	93.90	54.36	64.48	65.14	54.58	57.72	79.85	81.61	72.82	69.63	77.50	56.43
DUNet[25]	123.69	90.67	59.01	93.07	34.20	50.95	54.96	43.19	45.05	79.80	76.07	65.29	54.33	68.57	42.64
U-Net[26]	124.55	81.90	29.23	86.34	8.76	15.18	19.02	27.13	24.73	17.26	53.40	47.36	11.97	37.79	1.77
DANet[27]	198.00	92.70	68.81	93.69	47.69	66.05	70.18	53.01	56.15	77.73	82.89	72.24	72.18	77.87	56.06
PSPNet[28]	187.03	92.47	68.23	93.62	50.33	64.24	70.19	51.51	55.27	79.27	81.93	71.95	68.91	77.13	54.43
Trans4Trans-M[2]	34.38	**95.01**	**75.14**	**96.08**	55.81	**71.46**	69.25	**65.16**	**63.96**	83.84	**88.21**	**80.29**	76.33	**83.09**	**68.09**

al.[3] improved the model by adding a boundary feature extraction module and a glass reflection detecting module. Xie et al.[4] introduce a transformer-based network Trans2Seg with transformer encoder-decoder architecture. Zhang et al.[2] propose a semantic segmentation architecture Trans4Trans, shown in Table 1 that improves the architecture of Trans2Seg to achieve a better result. Lin et al.[5] present GlassSemNet that correlates the occurrence of glass surfaces with other surrounding objects (e.g., "windows" tend to occur with "curtains").

2.2 Methods using RGB-T

RGB-Thermal cameras contain infrared modules to detect thermal energy in addition to regular RGB cameras. Glass segmentation methods using RGB and thermal images take advantage of a sophisticated difference between visible light and thermal radiation. Glass seen in a daily environment is mostly silicate-based, which transmits visible light at a high rate. On the other hand, thermal radiation with wavelengths 8μm - 12μm cannot pass through silicate glass. Hence, glass surfaces invisible in RGB images will show in thermal images. This unique characteristic can greatly aid glass surface detection. Huo et al.[29] used RGB-Thermal image pairs to train a glass segmentation model and achieved improved results compared with previous methods.

2.3 Methods using RGB-D

RGB-D camera is a popular choice when it comes to robotic perception tasks. The extra depth camera can provide per-pixel depth information that assists object segmentation. However, the depth information acquired from transparent objects may be erroneous due to the violation of the Lambertian assumption. In the Lambertian assumption, a surface is expected to be equally bright from all viewing

angles; however, transparent surfaces both refract and reflect light, leading to two types of error shown in Fig. 4. Type I errors occur when light passes through the transparent material and reflects back from the background, causing an inaccurate depth estimation corresponding to the background depth. Type II errors occur due to specular reflections on the transparent surface, leading to missing depth information.

Some methods aim to fix the incorrect depth information. For example, ClearGrasp, presented by Sajjan et al.[30], uses surface normal/contact edge information and a global optimization algorithm to eliminate and refill the inaccurate depth information caused by transparent objects. Other methods take the presence of errors as a cue for glass surfaces. Lin et al.[31] propose a glass surface detection model that includes a Depth-missing Aware Attention (DAA) Module considering missing depth (Type II errors) regions as possible glass surfaces.

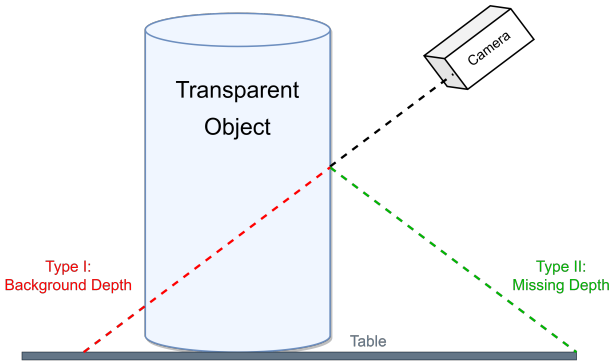


Figure 4: *Illustration of Transparent Object Depth Errors*

2.4 Methods using RGB-P

Polarization cameras are a particular type of camera that magnifies contrast between different polarized angles of light while filtering undesired reflections. The two parameters, DoLP and AoLP, which indicate the relative intensity and the orientation of the polarization axis, respectively, can be thought of as the material's intrinsic properties in that the polarized light is reflected. Therefore, the polarization light observations can be incredibly informative during processes of transparent object segmentation. Mei et al.[32] use such polarization methods as an additional cue along with the RGB image in their proposed model (PGSNet) to perform glass segmentation.

The methods above using additional features (thermal, depth, polarization) other than only RGB may have achieved reasonable results. Still, there are many disadvantages to these methods compared to RGB-only detections. First, additional sensors typically require higher costs. From purchase to maintenance, these sensors remain inaccessible to most researchers and developers from purchase to maintenance. Second, the non-portability and inconvenience of these sensors greatly limited their use in daily settings. Third, the time and effort needed to build a complex dataset usually leads to decreased quantity. **Meanwhile, regular RGB datasets can be gathered more rapidly and even more efficiently with synthetic data that can be prepared automatically. Considering all the factors, we used the simplest RGB image as input data**.

3 Dataset

In most computer vision object recognition and detection algorithms, there is a direct correlation between the amount of data and the model's performance. Glass segmentation and detection is a very niche area of study in semantic segmentation tasks with little current research. Earlier datasets like

GDD[1], proposed in 2020, only contained less than four thousand images. It mainly contained windows and glass walls. The first version of Trans10k[33] was proposed in the same year, containing the most comprehensive labeled data at 10,428 images of glass walls and other transparent objects. Still, only 5k of those images were dedicated to training. In 2022, TransTouch proposed using Blender to create CGI images on only small glass containers, which is more tailored to robotic arm applications, unlike indoor navigation. These datasets all require lots of manual work during collection and labeling. Segmentation tasks with more extensive datasets like the newly proposed SAM (Segment Anything)[34] contains 11 million images and over 1 billion masks. But even with a dataset of this scale, glass and transparent objects take up a tiny proportion of the dataset, resulting in undesirable performance when used for transparent object detection. Compared to segmentation datasets in other fields, current transparent object and glass segmentation datasets only have a limited size of ten thousand images. This paper introduces the Integrated Pipeline of Stable Diffusion Inpainting Dataset with Grounded-SAM shortened as Grounded Integrated Diffusion Inpainting Dataset **(Grounded-ID2)** to address the issues of limited data. Grounded-ID2 is partially displayed in Fig. 6

3.1 Transparent Object 3D Rendering Using Omniverse and Blender

Manually collecting data and analyzing them is always tedious. Before we thought of using AI technology for dataset generation, we explored the path of 3D rendering to simulate real-world environments. A dataset of 3D glass models. This dataset will be created by 3D modeling glass objects using Omniverse Suit. This will allow us to create a dataset of glass images with more accurate geometry than existing datasets. Omniverse Suit is a suite of tools for creating, editing and simulating 3D scenes. Blender is part of the Omniverse USD suite of tools. Blender is known for its accurate physics simulations, which allowed us to create 3D glass models with realistic geometry. Using Blender, a simulated dataset was produced manually. High dynamic range images (HDRI) were used to represent the background and lighting effects for the scenes shown in Fig. 5. The HDRI photo covers the entire area, offering a backdrop and complete natural lighting from all angles. The surroundings cover both indoor and outdoor environments.

Figure 5: *3D Rendered Scene using Blender*

We specifically included many glass walls and windows representative of a modern environment. Each scene contained up to 10 unrelated items randomly distributed throughout, rotated, and scaled. 54 different materials were used to render the scene. Realistic complex textures, displacements, and other characteristics representative of real-world environments are included.

However, generating these synthetic images was no easy task. It takes similar amounts of manpower to build these 3D scenes. Every object has to be adjusted and placed manually by hand. The scene needs to be designed first. Then, the materials needed are to be added accordingly. It is time-consuming and inefficient, just like manually collecting real-world data and labeling them. Therefore, we implement an automated method in the next section that can efficiently generate datasets for limitation scenarios.

3.2 Grounded-ID2: Grounded Integrated Diffusion Inpainting Dataset

The rise of AIGC (Artificial Intelligence Generated Content) has led to many use cases for the content generated. We thought of the limitation of datasets and the effectiveness and efficiencies of AIGC and created Grounded-ID2. This dataset contains a method of infinite image generation with automatic precise labeling, creating pixel-level accurate masks. Our method can relieve researchers from the tedious process of manual data collection and post-processing of labeling and mask creation. Saving tons of time and energy, it helps all researchers in the field dedicate more time to improving their models. Our method achieves this data generation process by using synthetic images and image segmentation techniques to cut out an area of an image for inpainting specifically. Inpainting is the task of reconstructing missing regions in an image, similar to image restoration. Stable Diffusion[35] enables high-quality inpaintings while maintaining the realistic elements and reasonable conceptions of an image. This method not only ensures the realistic looks of an image but also creates a mask simultaneously when the image is cut using segmentation models. Therefore, as shown in Fig. 6, this paper published Grounded-ID2, which comprised up to 20,000 (2,500 currently, with much more coming) pixel-accurate mask-labeled images. The next part will introduce the pipeline of Grounded-ID2 and the principles of each module.

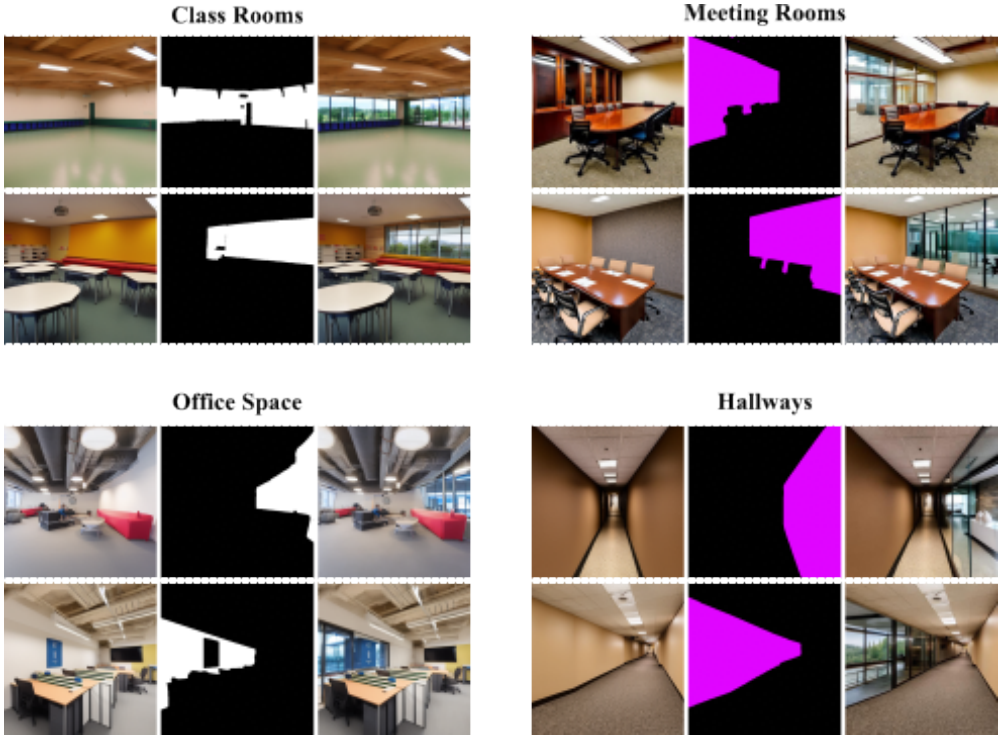


Figure 6: *Part of the Grounded-ID2 Dataset (More can be found in the Appendix)*

Algorithm 1 Image Processing Pipeline

Dependencies:
Import torch, PIL, cv2, numpy, os, sys
Imports: Import necessary modules and libraries
Load Models:
Load pre-trained models (Grounding DINO, SAM, Stable Diffusion)
Image Processing Loop:
for each image file in the directory **do**
    Load the image from the specified directory
    Detect objects within the image using Grounding DINO model
    if objects are detected **then**
        Segment the detected objects using the SAM
        Apply inpainting to the segmented objects using the Stable Diffusion method
        Save the original, mask, and inpainted images
        Increment the file counter
    end if
end for

3.2.1 Methods and Process Flow

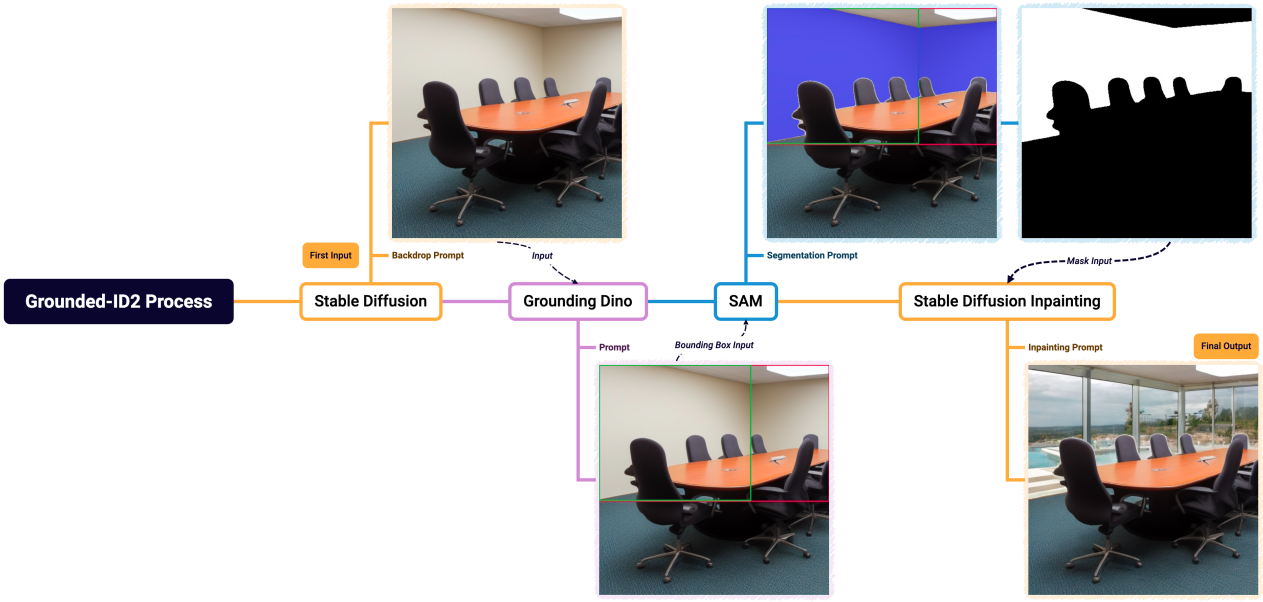


Figure 7: *The Framework for Grounded-ID2 Pipeline*

As shown in Fig. 7, our process takes a text input or a "Backdrop Prompt" to generate initial images with Stable Diffusion [35]. Then, the images will be fed to Grounding Dino[36] for object detection, and a bounding box will be generated, used to localize the area for mask creation and inpainting. The bounding box will be the input for SAM[34], which will output a desired pixel-accurate mask label. The mask will be used as input for stable diffusion[35] for inpainting, producing our desired data. This methodology is not only applicable to this specific task of generating glass data. This methodology can be generalized to any area of study because. Because current AIGC technologies are already compelling enough to simulate real data, we can use this advantage and mature segmentation task to create masks and labels for immature areas like glass segmentation. **Our methodology uses AIGC with segmentation technology to generate data that simulates the real world, performing segmentation on well-trained objects to create labels for developing tasks.**

3.2.2 Latent Diffusion

Current image generation models have evolved a long way. They can generate high-resolution images realistic enough to pass the Turing Test. Most people cannot easily discern between AI-generated images and actual photographs. This implies that these models are good enough to feed their generated images back into the training process of AI models. We use the state-of-the-art model, Stable Diffusion[35], which utilizes latent diffusion models for performance-efficient high-resolution image synthesis. Latent Diffusion Models are probabilistic models tailored to denoise variables with normal distribution to learn the pattern in the data distribution. That is similar to learning a fixed reverse Markov Chain. It uses a time-conditioned transformer based UNet[26] backbone built primarily with 2D convolutional layers, focusing on the most relevant bits.

We used Stable Diffusion v1.4 to generate up to 20,000 images applicable to different scenarios where transparent glass would appear indoors. Workspaces and meeting rooms are familiar places in modern office buildings that all have glass windows, glass walls, and glass doors. The images generated

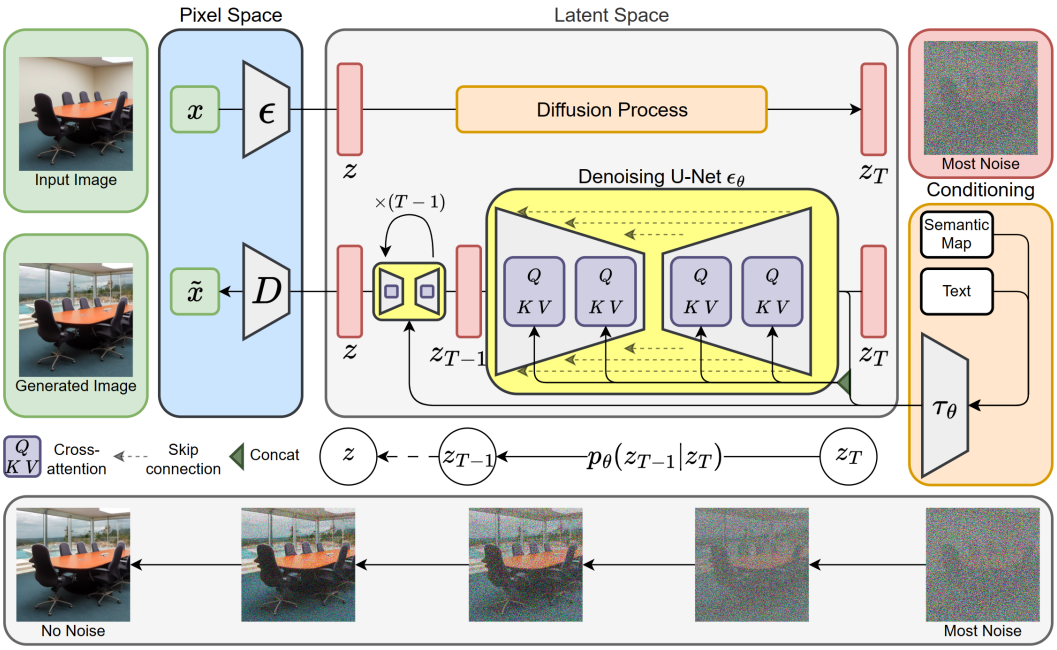


Figure 8: *Stable Diffusion Model. This diagram illustrates the inpainting process using Stable Diffusion to generate images for our dataset. The input image first undergoes a diffusion process, making the image noisy. Then, by adding the mask generated by SAM indicating the regions for inpainting and a conditioning prompt limiting the texture, the noisy image is denoised and outputted as the generated image.*

should not contain anything transparent or glass-related. There needs to be an area for a segmentation model to be cut out and prepared for inpainting, as shown in Fig. 7 These images would then be fed to a pre-trained Segment Anything model for segmentation. A pixel-accurate mask will be cut out, and the blank area will be left for inpainting.

3.2.3 Grounding Dino

For a given (Image, Text) combination, grounding DINO[36] produces numerous pairs of object boxes and noun phrases. Both object detection and REC (Referring expression comprehension) tasks can be coordinated with the pipeline. It concatenates all category names as input texts for object detection tasks in accordance with GLIP[37]. REC for each text input, a bounding box is necessary. The output for the REC is the output object with the highest scores.

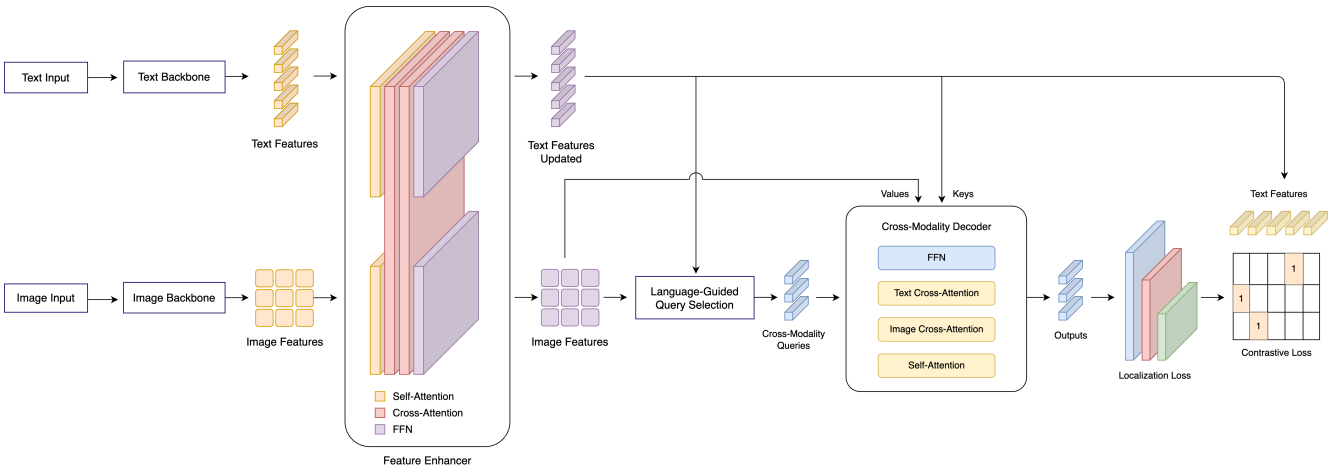


Figure 9: *DINO Model: takes Text and Image inputs with feature extraction and applies cross-attention*

The Dual-encoder, single-decoder design underlies grounding DINO. It has a cross-modality decoder for box refining, a language-guided query selection module, an image backbone for extracting image features, a text backbone for extracting text features, a feature enhancer for fusing image and text features, and a feature enhancer. In Fig. 9, the general framework is shown.

Grounding Dino initially uses an image backbone and a text backbone to extract vanilla text features and vanilla image features for each (Image, Text) combination. The two stock features are input into a feature enhancer module for cross-modality feature fusion. After acquiring cross-modality text and image features, we employ a language-guided query selection module to choose cross-modality queries from picture features. These cross-modality queries will be passed into a cross-modality decoder, much like the object queries in the majority of DETR-like models, to probe desired features from the two modal features and update themselves. The final decoder layer's output queries will be used to anticipate object boxes and extract phrases that go with them.

The extracted features from images and texts are first fed into a self-attention layer separately. Then, they are fed into two cross-attention layers of text-to-image and image-to-text, respectively, for cross-modality feature fusion. This creates a feature-enhancing layer to prepare for language-guided query selection. The final output query is then used for object bounding box extraction and description phrase extraction.

3.2.4 Segment Anything

Segment Anything[34] is a promptable segmentation model proposed by Meta, trained on 11 million images and over 1 billion masks. The image encoder uses an Masked Auto Encoder[38] pre-trained Vision Transformer (ViT)[39], shown in Fig. 10, efficiently used to process high-resolution image inputs.

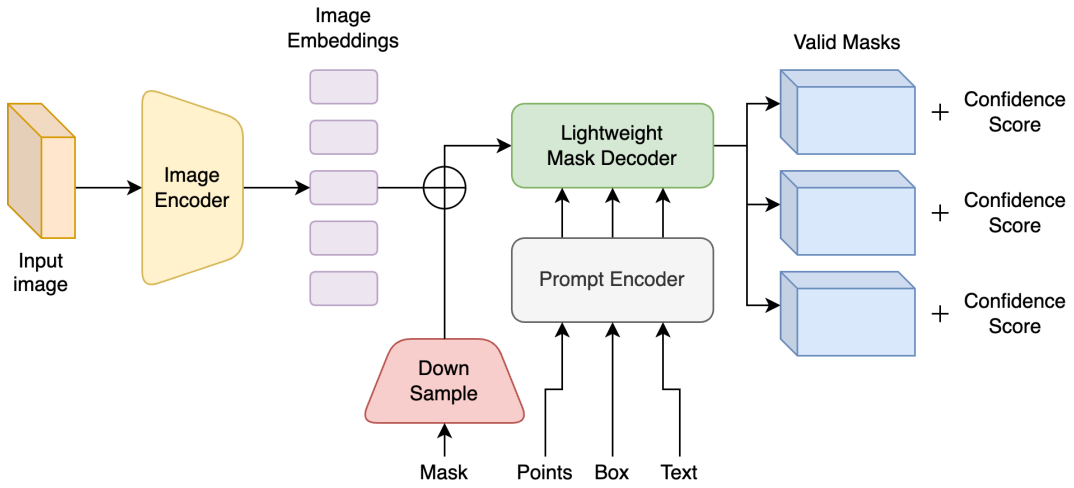


Figure 10: *Segment Anything Model (SAM) can take many different input prompts to create object masks*

The prompt encoder in Fig. 10 considers two sets of inputs: sparse inputs like points, boxes, texts, and masks. For each prompt type, points and boxes are identified by positional encodings added with learned embeddings. Free-form texts are represented with a directly implemented CLIP[40]. prompts (i.e., masks) are embedded using convolutional layers and summed element by element with the image embedding.

The mask decoder effectively converts the output token, prompt embeddings, and image embedding into a mask. This design uses a dynamic mask prediction head after a modified Transformer de-

coder block. Our updated decoder block employs prompt self-attention and cross-attention to update all embeddings in two ways (prompt-to-image embedding and vice versa). The image embedding is up-sampled after two blocks have been executed. An MLP calculates the mask foreground probability at each image position after mapping the output token to a dynamic linear classifier.

The model also takes into account ambiguous prompts. The model will average numerous valid masks into a single output when presented with one. In order to solve this, the model is modified to anticipate numerous output masks for a single prompt (see Fig. 6). We discovered Most typical scenarios can be addressed with three mask outputs (nested masks are frequently three deep: whole, part, and subpart). During training, it backprops the least amount of loss over masks. The model predicts a confidence score (i.e., predicted IoU) for each mask in order to rank them.

4 Trans3-Vision Model with DANN

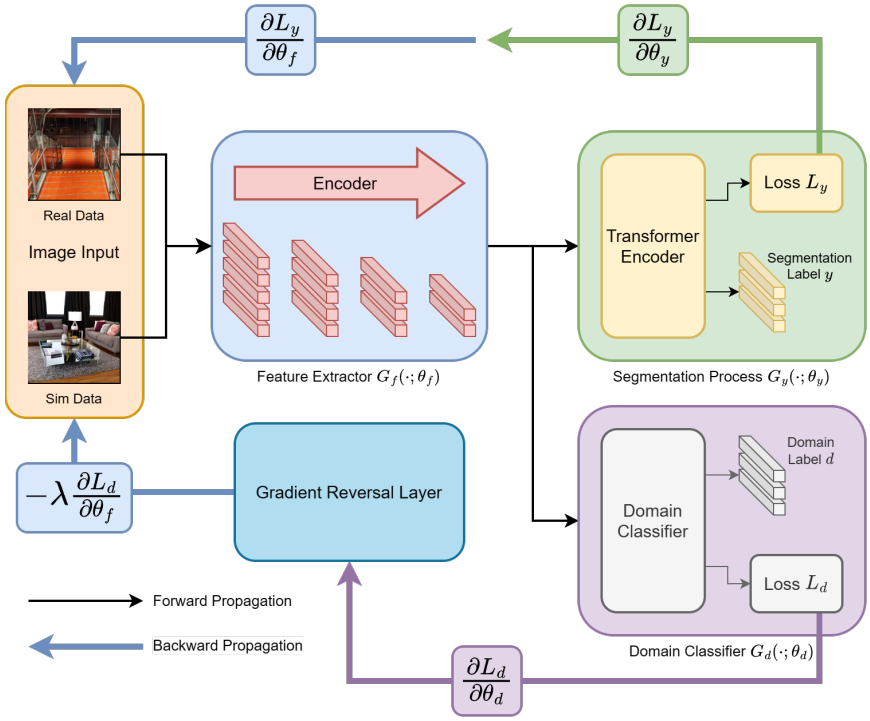


Figure 11: *Our Trans3-Vision utilizes Backpropagation and Gradient Reversal Layers*

In this part, we introduce Trans3-Vision, which is a DANN-based Transformer model that references the structure of Trans4Trans[2]. To ensure model accuracy and generalization, we use the DANN[41] method, the domain adaption allowed us to merge the real world and the generated world effectively. It easily combines two data sets of different characteristics while promising performance. Importantly, different features can be discovered from different datasets. This keeps Trans3-Vision lightweight while making it robust enough to avoid overfitting when tested in real-world situations. Our Trans3-Vision model is entirely made up of transformers and attention layers. This contrasts the ViT[39] transformer model, which has the advantage of gaining long-range dependencies. Shown in the middle of Fig. 11, The four-stage encoder is a PVT[42] adaptation. Furthermore, the transformer-based decoder is more reliable in parsing unforeseen data collected in the wild than CNN-based models that learn the inductive bias. However, a sizable dataset is needed to train a transformer model and we want to improve the generalization performance of our model. That is why we needed Grounded-ID2.

4.1 Vision Transformer Based Transparent Object Segmentation Backbone

Trans4Trans is the current top-performing model on the Trans10Kv2 dataset. As shown in Fig. 12, it consists of shared encoders and dual decoders. Heavily inspired by the ViT [39] transformer model, Trans4Trans's dual-head model is constructed solely with transformers. It constructed an efficient decoder shown in Fig. 12(c) using TPMs (Transformer Parsing Module) that contain only one attention layer, demanding fewer resources. The features F_1, F_2, F_3, F_4 from Fig. 12(a) are parsed by TPM modules. The features will be resized between stages and addition will be performed for feature synthesis. It uses 64 as the default channel number and sets the resolution of TPM to $\frac{H}{4} \times \frac{W}{4} \times C$.

We take inspiration from the Trans4Trans[2] model and ViT[39] transformer to develop our own Trans3-Vision with domain adaption.

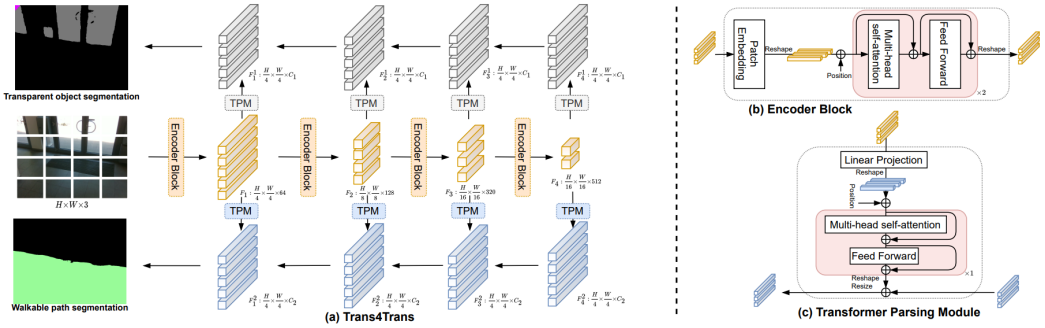


Figure 12: *The Trans4Trans model architecture[2]*

4.2 Implementing DANN (Unsupervised Domain Adaptation by Backpropagation)

DANN[41] (Domain Adaptation Neural Network) assumes that the model operates with input samples $x \in X$, where X denotes some input space, and with specific labels (output) y drawn from the label space Y. In the following context, it considers classification problems where Y constitutes a finite set ($Y = \{1, 2, \ldots, L\}$). Nevertheless, it is important to note that DANN's methodology exhibits generality and can accommodate any output label space that other deep feed-forward models can handle.

Additionally, it makes the assumption that two distributions, $S(x,y)$ and $T(x,y)$, exist on $X \otimes Y$. These distributions are commonly referred to as the source distribution and the target distribution (or the source domain and the target domain). Both of these distributions are characterized as intricate and remain unknown. Moreover, they exhibit similarity but distinctiveness, implying that S experiences a certain domain shift about T.

The ultimate objective is to predict labels y based on input x for the target distribution. During the training phase, it has access to a substantial set of training samples denoted as $\{x_1, x_2, \ldots, x_N\}$ originating from both the source and target domains, following marginal distributions $S(x)$ and $T(x)$, respectively. To distinguish between the two domains, it employs binary variables d_i (referred to as domain labels) for each example. These variables indicate whether x_i originates from the source distribution ($x_i \sim S(x)$ if $d_i = 0$) or from the target distribution ($x_i \sim T(x)$ if $d_i = 1$). For instances derived from the source distribution ($d_i = 0$), it possesses knowledge of the corresponding labels $y_i \in Y$ during the training phase. Conversely, we lack label information during training for instances arising from the target domains and aim to predict these labels during testing.

During training with Trans3-Vision, the network takes as input the source domain dataset with image

classification labels and the target domain dataset without image classification labels, as well as domain classification labels for both source and target domain data. In other words, the source domain dataset has information about image classification labels, but the target domain dataset does not.

5 Experiments

5.1 Experiment Environment

We implement Grounded-ID2 with Python 3.9, PyTorch 2.0.1, Cuda 11.7, and cudnn 8.5. The experiment was conducted on a rented server equipped with an Intel Zeon Gold 5218R CPU with dual Quadro RTX 6000 GPUs and 128GB of RAM. We use these environments to test and train our Trans3-Vision model and Grounded-ID2 pipeline. The tran3-Vision model will be deployed on an Nvidia Jetson AGX Orin developer platform.

5.2 Evaluation Metrics

To quantitatively evaluate the performance of the proposed model, we adopted two metrics commonly used in image segmentation fields. The first metric, Intersection over Union (IoU) $= \frac{\text{Area of Overlap}}{\text{Area of Union}}$, measures the amount of overlap between two bounding boxes—a predicted bounding box and a ground truth bounding box. It is calculated by the ratio of the intersection of the two boxes' areas to their combined areas. The second metric, Pixel Accuracy (ACC), represents the number of correctly predicted pixels, compared to the total number of predicted pixels. For both metrics, the higher their value, the better the results are.

5.3 Experiment Methodology

To measure the effectiveness of our generated data, we first created a baseline by evaluating Trans4Trans[2] on the state-of-the-art Trans10kv2 [33] test dataset. The baseline will be used to set a standard and compare all other test results. An experiment is performed on our generated Grounded-ID2 without any training adjustments or tuning. Then, we implement Grounded-ID2 directly into the training of the model. We compare the model results when only trained on the Trans10Kv2 dataset and the results when our augmented training data is added to the pool. We experimented with how different proportions of our data and Trans10Kv2 data could affect the results. We further perform real-world tests by capturing real-world scenarios with an RGB camera.

Figure 13: *Trans10Kv2 Color Palette*

5.3.1 Evaluating Trans4Trans on the Trans10Kv2 Dataset Only

This section presents the test results to evaluate previous SOTA (state-of-the-art) model's performance on the Trans10Kv2 test dataset and the real-world dataset we collected. **Even though the previous SOTA models like Trans4Trans achieved excellent results (see Table 2) on the Trans10Kv2 test dataset, we can see from Fig.15 that the real-world performance is not ideal**, making incorrect segmentation in many areas. The results imply that Trans4Trans lacks generalization and is not robust against different scenarios and applications. This might be because of how most image on the Trans10Kv2 dataset looks. From Fig.14 's visualization, we can see that most glass regions have high contrast borders, which may greatly influence test results. These high contrast borders are formed by low light to bright light environments or just an effect due to different colored paint.

Figure 14: *Results on the Trans10Kv2 test dataset*

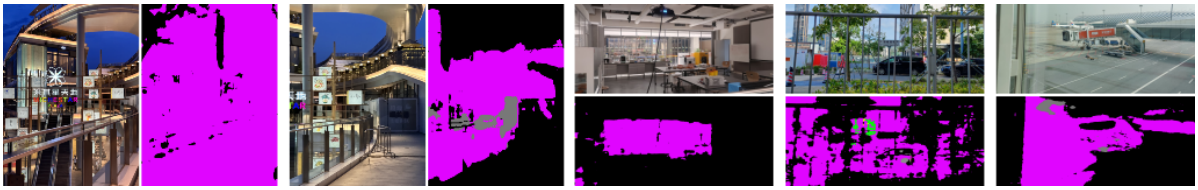


Figure 15: *Results on our real-world data*

Table 2 shows the IoU score obtained using the Trans4Trans model on the different categories of the Trans10kv2 dataset. We can see the model has a varying performance from 55% mIoU to 86% mIoU on glass object segmentation from different Trans10kv2 categories. The average mIoU score is 74.586%. These scores are used as a baseline for further improvements and comparisons.

5.3.2 Evaluating Trans4Trans on Grounded-ID2 without Training

Now, we test to see how Trans4trans performs without training with any of the Grounded-ID2 data on the Grounded-ID2 test dataset. Fig. 16 and Table 3 show that performance drops significantly. From Fig. 16, we can see that the first image had an incorrect segmentation while others are either incomplete or unconfident, and some even detected the wrong category. This again proves previous SOTA models like Trans4Trans have very bad generalizations and robustness.

To address this problem, we combine our Grounded-ID2 dataset with the Trans10Kv2 dataset to train the model. This will allow the model to learn from more scenarios and aim to improve overall generalization and robustness.

Table 3, is evident that only training on the Trans10kv2 does not allow the model to perform ideally

Table 2: *The Baseline result on Trans10K v2 datasets*

Class Name	**Background [0]**	**Shelf [1]**	**Jar or Tank [2]**	**Freezer [3]**	**Window [4]**
IoU (%)	96.01	55.74	70.79	68.30	63.62
Class Name	**Glass Door [5]**	**Eyeglass [6]**	**Cup [7]**	**Floor Glass [8]**	**Glass Bow [9]**
IoU (%)	63.50	83.10	86.76	79.92	74.79
Class Name	**Water Bottle [10]**	**Storage Box [11]**	**Total Mean**	**PixAcc**	94.88
IoU (%)	83.67	68.84		**mIoU**	74.59

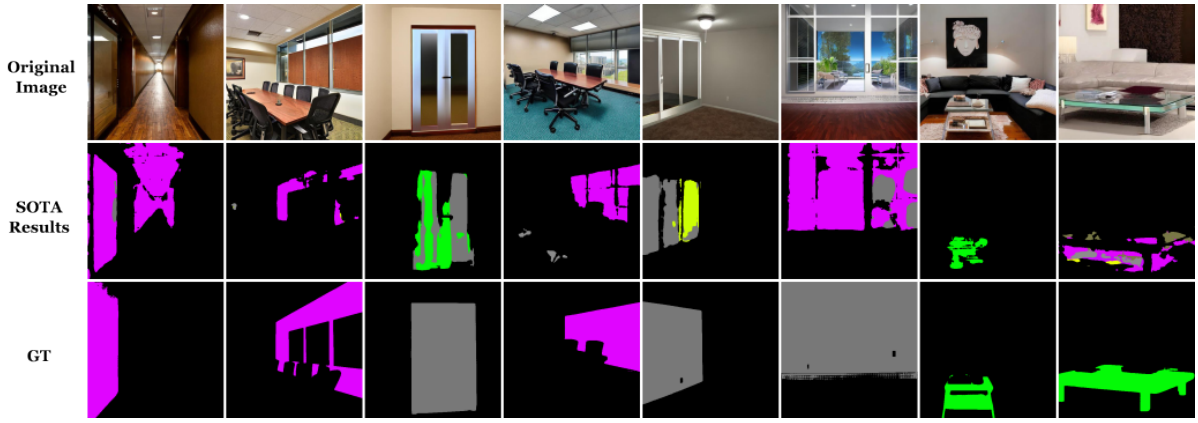


Figure 16: *Results on our Grounded-ID2 without training*

Table 3: *Comparison of baseline performance with initial performance on Grounded-ID2 dataset*

Class ID	Class Name	Grounded-ID2 IoU (%)	Trans10kv2 IoU (%)
0	Background	86.58	96.01
4	Window	0.64	63.62
5	Glass Door	43.43	63.50
8	Floor Glass	30.07	79.92
11	Storage Box	4.26	68.84
-	**Average**	**32.57**	**75.76**

on the proposed Grounded-ID2 dataset. These results show that if the proposed model represents the real world, then Grounded-ID2 is undoubtedly a great asset to existing glass segmentation datasets. To confirm how representative of real-world environments Grounded-ID2 is, an experiment is conducted for the performance of the model after training with Grounded-ID2.

5.3.3 Training with Grounded-ID2 and Trans10Kv2 Combined

For this experiment, we experiment with different datasets constructed of different proportions from Grounded-ID2 and Trans10Kv2. The training process is shown in Tabel 4 and Fig. 17. First, we only train our model (The green line ”1to0” on Fig. 17) on grounded-ID2 and evaluate its performance,

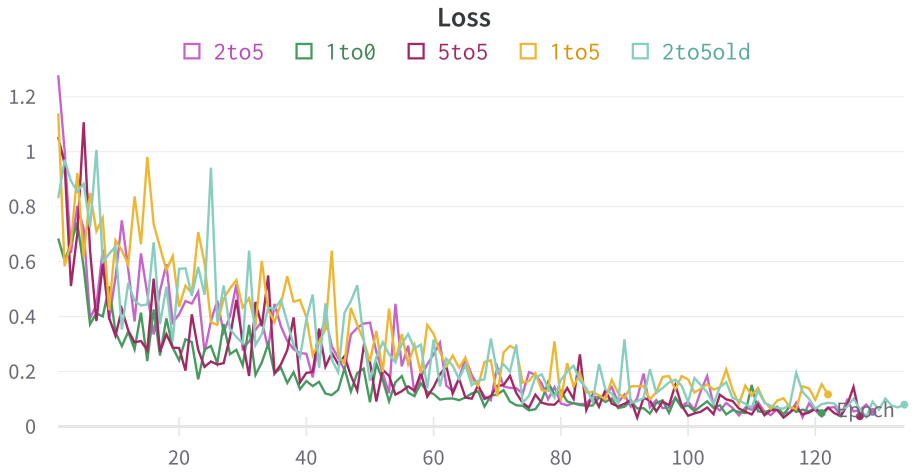


Figure 17: *Training Loss of different datasets with different proportions*

resulting in a lackluster result of 49% mIoU, which is still a 55% increase in mIoU score over the baseline. This presents room for improvement in our future research. Our current dataset is only meant to be a handy asset but cannot be used solely on its own. Second, Grounded-ID2 is downsampled to match the 1 to 5 ratio. There is a minor increase on our test set, but it also dropped more on the Trans10Kv2 test set. The performance on the Grounded-ID2 test is able to increase because it trained on a larger dataset with a wider range of data. However, we see a dramatic decrease for the Trans10Kv2 test set compared to the baseline. This is reasonable because they had higher performance metrics but they have poor generalization and robustness against new scenarios. There's the likelihood of overfitting on the Trans10Kv2 dataset, causing any introduction of new data to break the line.

With further experiments, the 2 to 5 ratio has the best results, but it is still not ideal for real applications. To improve this problem, we used a solution of domain adaption introduced in section (Sec 4.1).

Table 4: *Training the model with different proportions of Grounded-ID2 to Trans10Kv2*

Data Proportion	Dataset	Background	Window	Glass Door	Floor Glass	Storage Box	Average
Baseline	Grounded-ID2	86.58	0.64	43.43	30.07	4.26	33.00
	Trans10kv2	96.01	63.62	63.50	79.92	68.84	**74.38**
1 to 0	Grounded-ID2	88.79	13.87	70.69	43.21	30.19	49.35
	Trans10kv2	86.24	37.01	30.18	50.85	9.78	42.81
1 to 5	Grounded-ID2	90.35	16.62	77.52	46.69	21.74	50.58
	Trans10kv2	85.99	36.01	29.93	50.32	6.12	41.67
2 to 5	Grounded-ID2	91.85	9.66	78.39	54.99	40.67	**55.11**
	Trans10kv2	85.62	36.45	31.53	52.76	12.16	43.70
5 to 5	Grounded-ID2	91.54	9.41	77.70	55.86	33.72	53.65
	Trans10kv2	85.79	37.01	32.13	51.52	11.02	43.49

5.3.4 Training Tran3-Vision with DANN Implemented

Here, we present Trans3-Vision results after implementing DANN[41]. The top section of Fig. 18 shows visualizations from our collected real-world data. The 4th picture from the left on the first row exemplifies great performance under complex light conditions. The right-most image shows how Trans3-Vision deals with double layers of glass. There's a glass partition close to the POV (Point of View) on the right of the image and glass sections at the further end. This image shows Trans3-Vision's understanding of dimension in the image and was able to correctly segment glass panes close up and in low light conditions. From the Trans10Kv2 test set in the middle of Fig. 18, we can see improved categorization and segmentation accuracy performance. Our Grounded-ID2 test dataset displays similar performance as the other datasets.

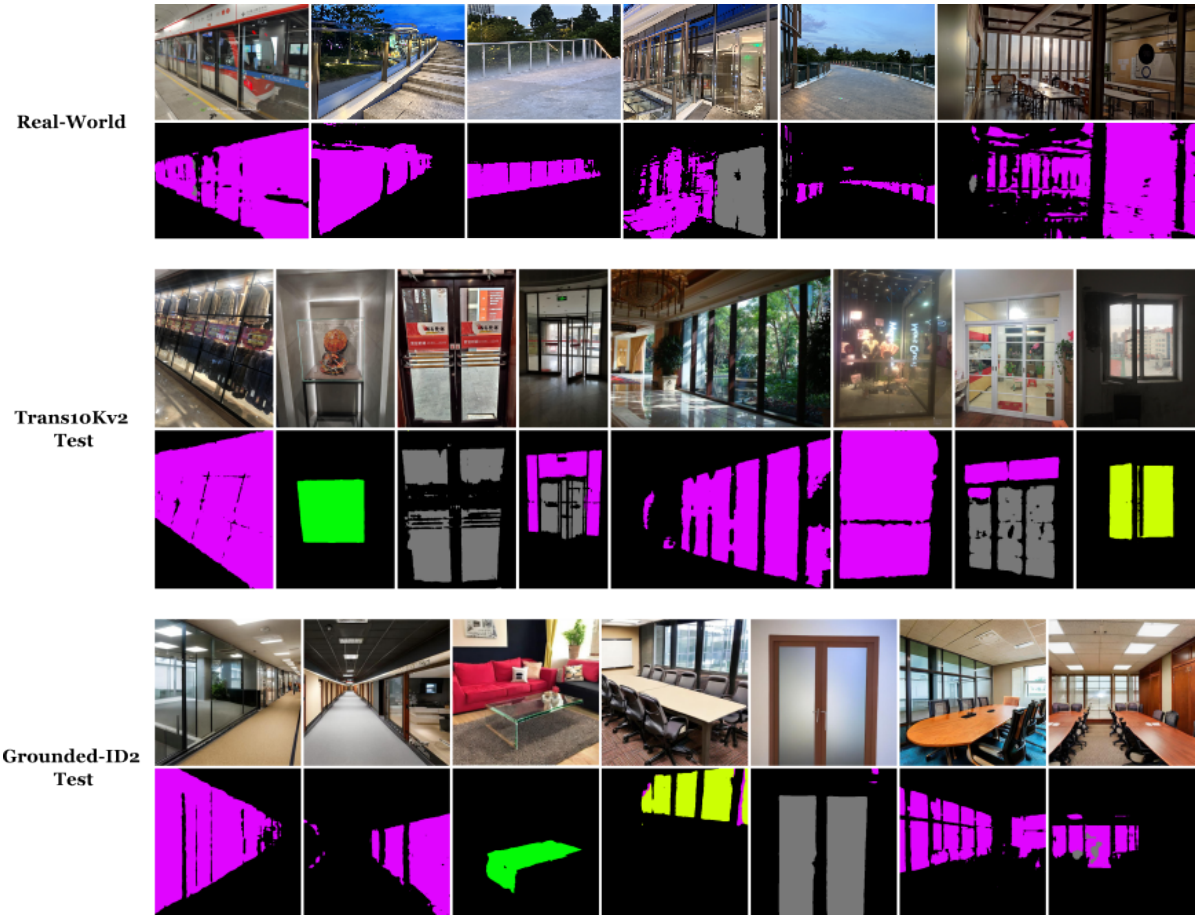


Figure 18: *Trans3-Vision results between all three datasets*

In addition to result visualizations, we see in Table 5, there's a drastic improvement in performance for both models. We used the best data proportions of 2 to 5 to train Trans3-Vision. Our solution effectively addresses the problem of overfitting to an unbalanced dataset of heavy-weight generated data. Results on Table 5 show a 2.1% increase over the baseline Trans10Kv2 test set and a 69.8% increase over the baseline of our Grounded-ID2 test set.

Table 5: *Trans3-Vision Results*

Data Proportion	Dataset	Background	Window	Glass Door	Floor Glass	Storage Box	Average
Baseline	Grounded-ID2	86.58	0.64	43.43	30.07	4.26	33.00
	Trans10kv2	96.01	63.62	63.50	79.92	68.84	74.38
Trans3-Vision	Grounded-ID2	91.97	13.23	77.14	55.68	42.16	**56.04** (+69.8%)
	Trans10kv2	96.03	65.51	67.17	83.02	67.98	**75.94** (+2.1%)

5.3.5 Disrupting the Data Pool to Improve Performance

Another method we tried is to disrupt the data pool, involving the addition of irrelevant or noisy data, to improve training results in machine learning. We added invalid images containing no glass or irrelevant objects. We hope models will become more robust by intentionally introducing variability and noise, reducing the risk of overfitting. However, no sign of improvement was shown after much experimentation with this method.

6 Conclusion

This project originated from the endeavor to build an autonomous driving vehicle on our campus, with the aim of addressing the challenge of transparent object recognition. Throughout the research, several key findings and developments have emerged:

AI-Assisted Dataset Generation: To overcome the scarcity of available datasets, We ultimately developed Grounded-ID2, an automatic pixel-accurate data generation pipeline applicable not only to glass detection. This approach proved effective in augmenting the dataset and overcoming the limitations of manual data collection.

Trans3-Vision Neural Network: In order to bridge the gap between virtual training environments and real-world applications, we designed the Trans3-Vision neural network, improving our training methods and achieving ideal results performing on real-world data with a 2.1% mIoU score increase compared to other state-of-the-art models. This transfer learning model facilitates a seamless transition of autonomous vehicles from virtual environments to real-world situations, enhancing their generalization performance.

Overall, this research highlights the significance of AI-assisted dataset generation and transfer learning approaches in addressing the challenges associated with transparent object recognition in autonomous driving. Moving forward, our future plans entail evaluating the vehicle's performance in various challenging environments and assessing its ability to adapt to different circumstances. The findings emphasize the positive impact of AI-generated datasets on training neural networks to handle complex scenarios effectively. We have already open-sourced our dataset and will continuously improve the AI-generated datasets and their alignment with real-world scenarios.

References

[1] H. Mei, X. Yang, Y. Wang, Y. Liu, S. He, Q. Zhang, X. Wei, and R. W. Lau, "Don't hit me! glass detection in real-world scenes," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3684–3693, 2020.

[2] J. Zhang, K. Yang, A. Constantinescu, K. Peng, K. Müller, and R. Stiefelhagen, "Trans4trans: Efficient transformer for transparent object and semantic scene segmentation in real-world navigation assistance," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[3] J. Lin, Z. He, and R. W. Lau, "Rich context aggregation with reflection prior for glass surface detection," in *Proc. CVPR*, 2021.

[4] E. Xie, W. Wang, W. Wang, P. Sun, H. Xu, D. Liang, and P. Luo, "Segmenting transparent object in the wild with transformer," *arXiv preprint arXiv:2101.08461*, 2021.

[5] J. Lin, Y.-H. Yeung, and R. W. Lau, "Exploiting semantic relations for glass surface detection," *NeurIPS*, 2022.

[6] M. Liu and H. Yin, "Feature pyramid encoding network for real-time semantic segmentation," *arXiv preprint arXiv:1909.08599*, 2019.

[7] S. Mehta, M. Rastegari, L. Shapiro, and H. Hajishirzi, "Espnetv2: A light-weight, power efficient, and general purpose convolutional neural network," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9190–9200, 2019.

[8] R. P. Poudel, U. Bonde, S. Liwicki, and C. Zach, "Contextnet: Exploring context and detail for semantic segmentation in real-time," *arXiv preprint arXiv:1805.04554*, 2018.

[9] R. P. Poudel, S. Liwicki, and R. Cipolla, "Fast-scnn: Fast semantic segmentation network," *arXiv preprint arXiv:1902.04502*, 2019.

[10] H. Li, P. Xiong, H. Fan, and J. Sun, "Dfanet: Deep feature aggregation for real-time semantic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9522–9531, 2019.

[11] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "Enet: A deep neural network architecture for real-time semantic segmentation," *arXiv preprint arXiv:1606.02147*, 2016.

[12] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018.

[13] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 10, pp. 3349–3364, 2020.

[14] P. Chao, C.-Y. Kao, Y.-S. Ruan, C.-H. Huang, and Y.-L. Lin, "Hardnet: A low memory traffic network," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 3552–3561, 2019.

[15] G. Li, I. Yun, J. Kim, and J. Kim, "Dabnet: Depth-wise asymmetric bottleneck for real-time semantic segmentation," *arXiv preprint arXiv:1907.11357*, 2019.

[16] Y. Wang, Q. Zhou, J. Liu, J. Xiong, G. Gao, X. Wu, and L. J. Latecki, "Lednet: A lightweight encoder-decoder network for real-time semantic segmentation," in *2019 IEEE international conference on image processing (ICIP)*, pp. 1860–1864, IEEE, 2019.

[17] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "Icnet for real-time semantic segmentation on high-resolution images," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 405–420, 2018.

[18] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 325–341, 2018.

[19] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang, "Denseaspp for semantic segmentation in street scenes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3684–3692, 2018.

[20] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 801–818, 2018.

[21] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

[22] Y. Yuan, L. Huang, J. Guo, C. Zhang, X. Chen, and J. Wang, "Ocnet: Object context for semantic segmentation," *International Journal of Computer Vision*, vol. 129, no. 8, pp. 2375–2398, 2021.

[23] G. Lin, A. Milan, C. Shen, and I. Reid, "Refinenet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1925–1934, 2017.

[24] E. Xie, W. Wang, W. Wang, M. Ding, C. Shen, and P. Luo, "Segmenting transparent objects in the wild," in *Computer Vision – ECCV 2020* (A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds.), (Cham), pp. 696–711, Springer International Publishing, 2020.

[25] Q. Jin, Z. Meng, T. D. Pham, Q. Chen, L. Wei, and R. Su, "Dunet: A deformable network for retinal vessel segmentation," *Knowledge-Based Systems*, vol. 178, pp. 149–162, 2019.

[26] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pp. 234–241, Springer, 2015.

[27] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[28] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[29] D. Huo, J. Wang, Y. Qian, and Y.-H. Yang, "Glass segmentation with rgb-thermal image pairs," *IEEE Transactions on Image Processing*, vol. 32, pp. 1911–1926, 2023.

[30] S. S. Sajjan, M. Moore, M. Pan, G. Nagaraja, J. Lee, A. Zeng, and S. Song, "Cleargrasp: 3d shape estimation of transparent objects for manipulation," 2019.

[31] J. Lin, Y. H. Yeung, and R. W. H. Lau, "Depth-aware glass surface detection with cross-modal context mining," 2022.

[32] H. Mei, B. Dong, W. Dong, J. Yang, S.-H. Baek, F. Heide, P. Peers, X. Wei, and X. Yang, "Glass segmentation using intensity and spectral polarization cues," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12612–12621, 2022.

[33] E. Xie, W. Wang, W. Wang, M. Ding, C. Shen, and P. Luo, "Segmenting transparent objects in the wild," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pp. 696–711, Springer, 2020.

[34] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," 2023.

[35] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," 2022.

[36] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu, and L. Zhang, "Grounding dino: Marrying dino with grounded pre-training for open-set object detection," 2023.

[37] L. H. Li, P. Zhang, H. Zhang, J. Yang, C. Li, Y. Zhong, L. Wang, L. Yuan, L. Zhang, J.-N. Hwang, *et al.*, "Grounded language-image pre-training," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10965–10975, 2022.

[38] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," 2021.

[39] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021.

[40] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," 2021.

[41] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proceedings of the 32nd International Conference on Machine Learning* (F. Bach and D. Blei, eds.), vol. 37 of *Proceedings of Machine Learning Research*, (Lille, France), pp. 1180–1189, PMLR, 07–09 Jul 2015.

[42] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," 2021.

Acknowledgments

This research project would not have been possible without the support of the following people.

We want to sincerely thank the teachers at our current school, Mr. Yang and Ms. Yi, for supporting this long-term project and guiding us through this research.

We want to express our sincere gratitude to Professor Tan Ping[1] at Hong Kong University of Science and Technology, Department of Electronic & Computer Engineering. He has over 13k citations on Google Scholar and was previously the head of XR Lab at DAMO Academy, Alibaba. We gratefully thank him for taking time out of his busy schedule to answer all of our questions and helping us avoid unnecessary mistakes during our learning and research. We thank him for all the advice he offered throughout our project. Our knowledge of the subject greatly expanded under Professor Tan's guidance.

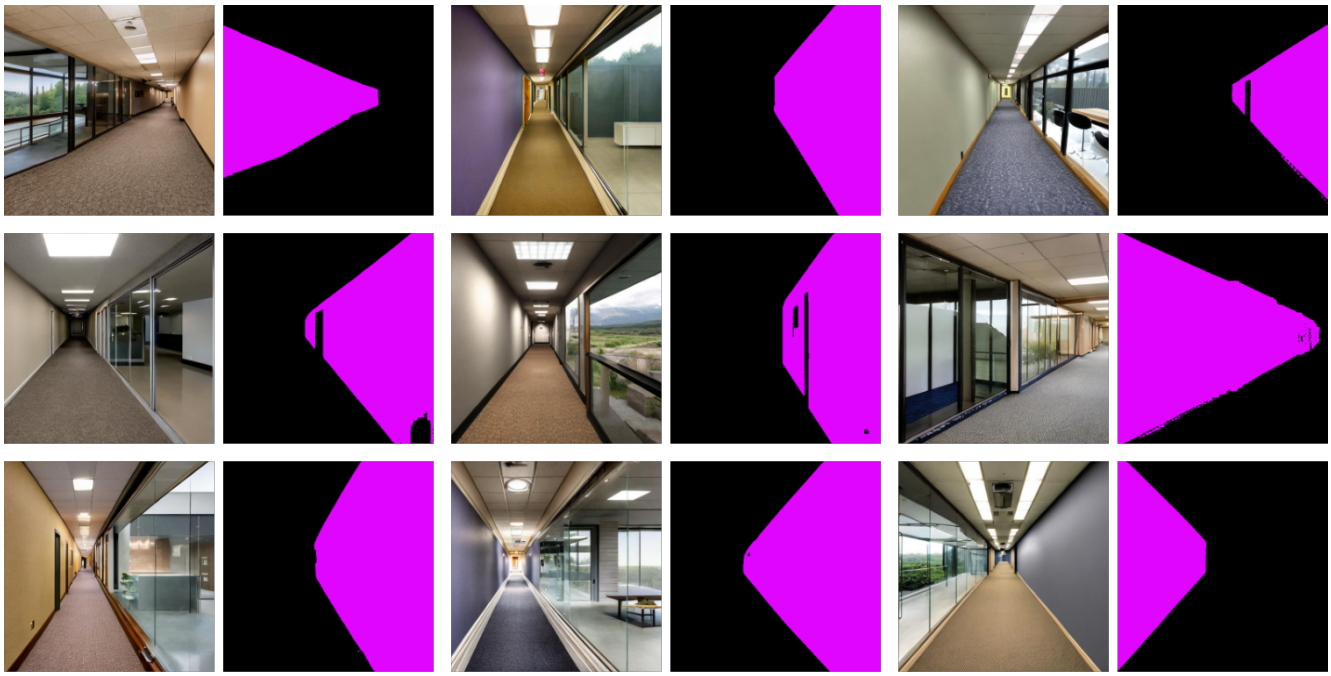
During the project of the robotic car and the Trans3-Vision and Grounded-ID2, the equipment is funded by our teachers and parents. We evenly split the bill to buy the robot with the ownership belonging to our school. Our training devices are rented at Tencent Cloud. Junyang Wang and Zhongshu Liu worked as a team to finish this research together. Both of us were responsible for the literature review, project design, and research paper. Junyang Wang was responsible for Grounded-ID2 datasets design and construction and Zhongshu Liu was responsible for idea proposal and data analysis and performing experiments. Again, we are very grateful to our parents and teachers who gave their utmost support.

We would also like to thank our friend Louis Hu for offering us a peaceful workspace. Sophia Li is a classmate who kindly helped us improve one of our graphical assets. Vicky Tan also helped us overcome challenges by supporting us along the way with companionship, alleviating our pressure, and helping us forge on.

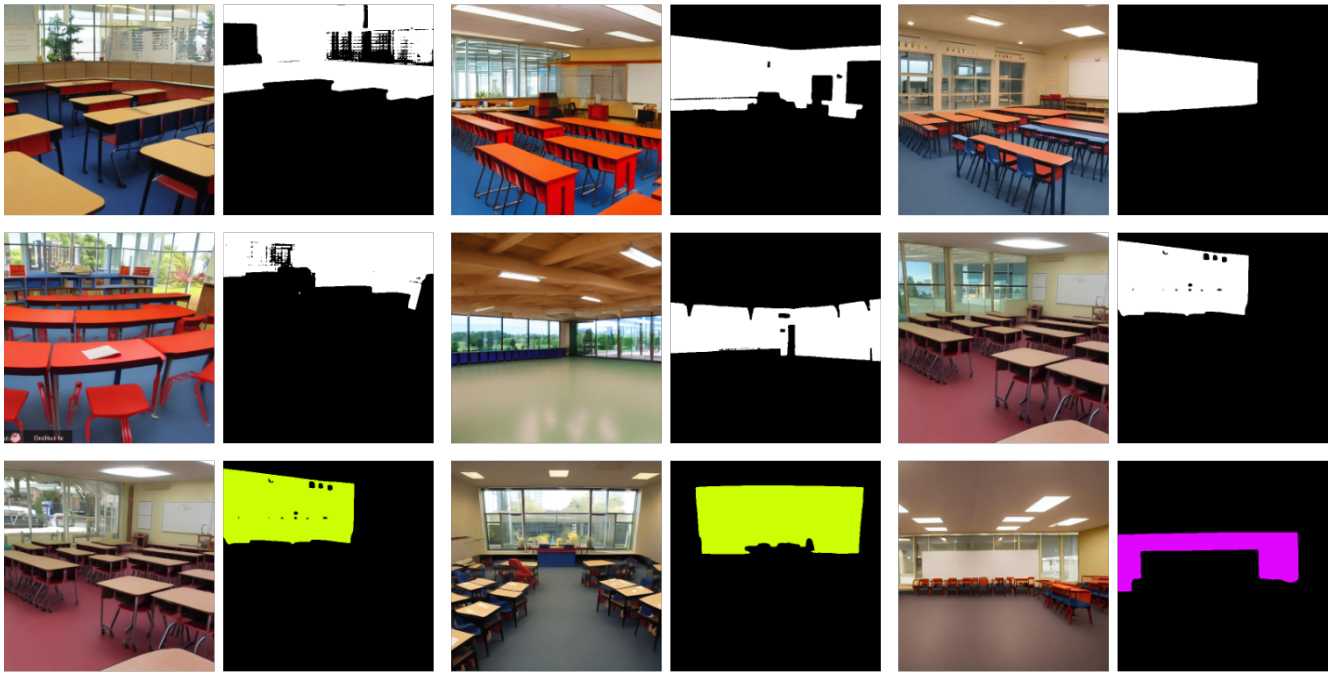
Last but not least, we want to thank everyone in the industry for publishing their research and making their research open-source. Without them, we would not have been able to build our autonomous car or continue with this research.

[1] https://ece.hkust.edu.hk/pingtan

Appendix: Visualization of Grounded-ID2

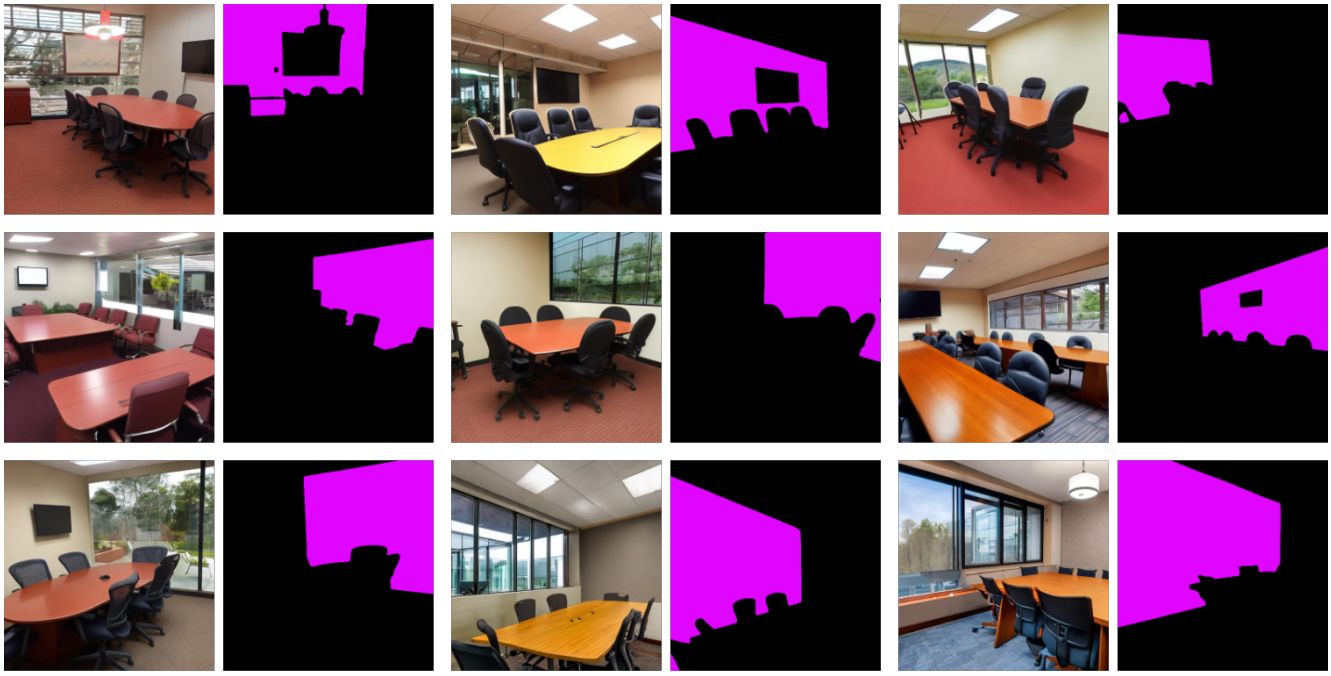


Hallway



Classroom

Figure 19: *The Hallway and Classroom classes from Grounded-ID2*



Conference Room

Living Room

Figure 20: *The Conference Room and Living Room classes from Grounded-ID2*

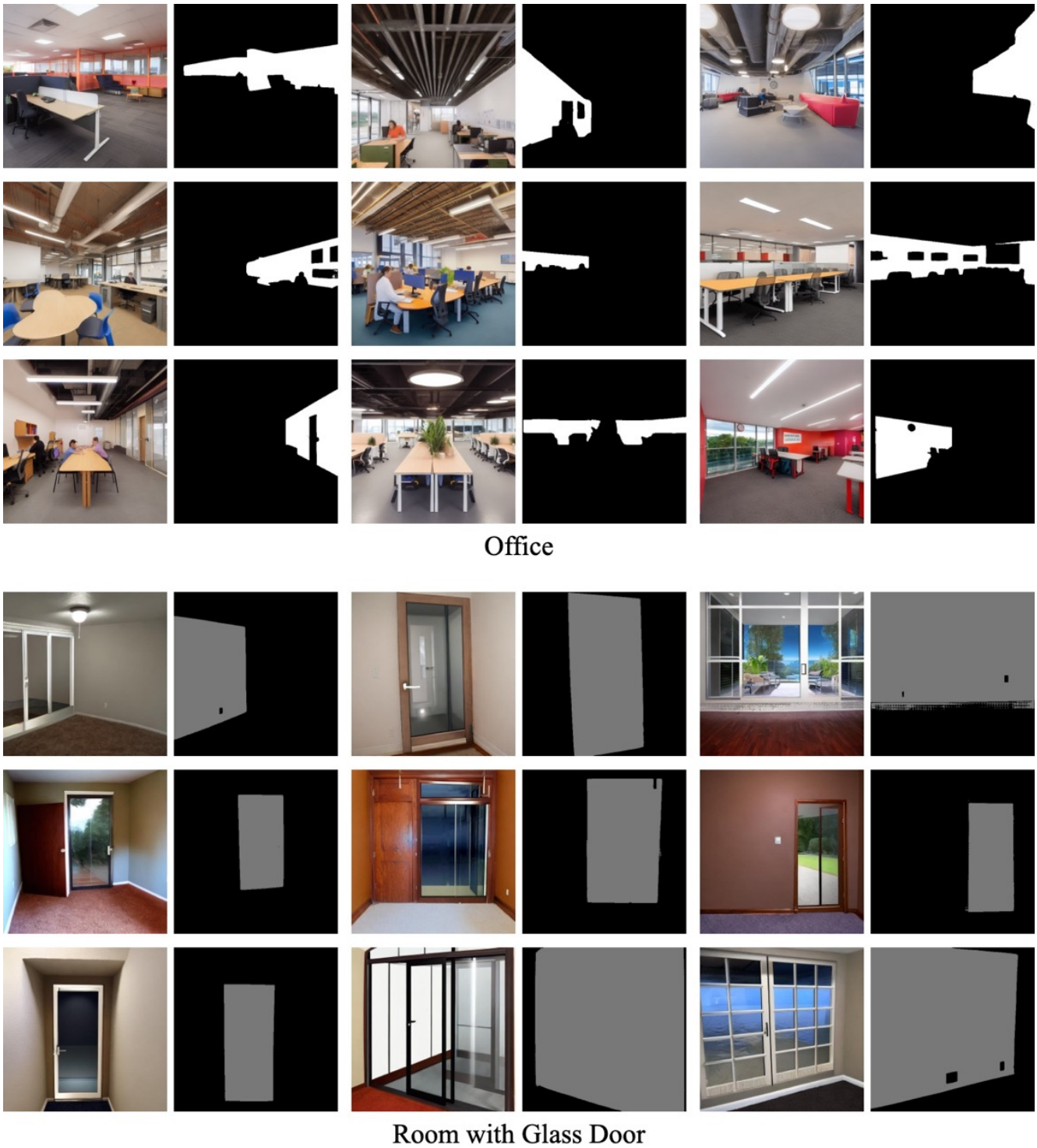
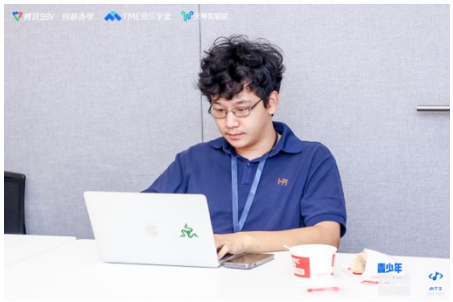


Figure 21: *The Office and Room with Glass Door classes from Grounded-ID2*

王郡阳 MICHAEL WANG

The Affiliated High School of South China Normal University
+86 18578659398 wjy.michael2022@gdhfi.com

I studied in US when I was 8 y/o in 3rd grade and relocated back to China in 9th grade. I love all new technology and devoted all my spare time to gaining knowledge on different computers and i's applications. I also love music and try to discover ways to improve music using technology like my recent experience at Tencent.



HONORS & ACADEMICS

SAT score: 1550 / 1600 Top 1% among all testers worldwide	*Aug 2023*
TOFEL iBT score: 110/120	*Sept 2022*
International Academic Review of The Year Hi World! Scholar of the Year (Highest Honor)	*Mar 2022*
National Championship of GFSSM Future Space Messenger Conference Competition	*Dec 2021*
HiMCM Finalist Gold Award / NCTM Winner Award (two teams only worldwide)	*Nov 2021*
ASDAN Business Simulation Gold Award with the Highest Honor	*Nov 2021*

SELECTED PUBLICATIONS

"A Novel Machine Learning Algorithm: Music Arrangement and Timbre Transfer System" was published at the ICICSP conference and was retrieved by IEEE and Scopus — *Jun 2022*

Junyang Wang, Wanzhen Sun, Rubi Wu, Yixuan Fang, Ruibin Liu. et. al.

ACTIVITIES & PROJECTS

Hardware research and development of autonomous vehicle	*Jun 2021 ~ current*
Summer Program at Tencent Ethereal Audio Lab studying AI Music & Audio Rating	*Jul 2023 - Aug 2023*
Participated in YOC competitions and won Honourable Mentions	*Apr 2022 ~ Nov 2022*
Created AP Calculus Club as club leader	*Apr 2022*

ABILITIES & STRENGTHS

Language: Native level in both English and Mandarin; beginning level in Spanish and Latin

Tech skills: Knowledgeable on AI and avid Python programmer, huge tech fan

Specialties: Intermediate Saxophone player; knowledgeable on computer and high-tech

刘忠恕 Zhongshu Liu

The Affiliated High School of South China Normal University

+86 15818870292

liuzs.james2022@gdhfi.com

Introduction

Growing up, I actively engage with the process of exploring the world around me. For example, I am sincerely fascinated by the vague and unknown. On one part, this leads my passion towards theoretical physics and astrophysics. On the other, this contributes to my interest in undefined forms of art, such as music.

Educational Background

- Academic Excellence Scholarship G10 (Top 10%)
- Zeal Labs 2023 Summer Program (Astrophysics): Research on Changing-Look Events in Active Galactic Nuclei (Final Grade: A)
- AwesomeMath 2022 Summer Program: Math Counts with Proofs - Level 1
- Y10 GPA: 4.98/5.00
- AP Calculus BC: 5
- SAT score: 1540/1600 (Top 1% of all testers)
- TOEFL iBT score: 115/120

Academic Honors

- Gold Award, BPhO Senior Challenge 2022
- Honor Roll, Top 5%, AMC10 2022

Interests

- Learned piano and Guzheng for 8 years, passed ABRSM 8th grade and National Music Association 10/10 grade, music performance on CCTV music and various school activities, posts original and cover songs on personal social media account.
- Current school Acapella club president, organizes and performs multiple self-arranged songs in various school performances.