The FineView Dataset:
A 3D Scanned Multi-view Object Dataset of Fine-grained Category Instances

Suguru Onda
Brigham Young University
ondas@byu.edu

Ryan Farrell
Brigham Young University
farrell@cs.byu.edu

Abstract

In the past decade state-of-the-art deep learning models have shown impressive performance in many computer vision tasks by learning from large and diverse image datasets. Most of these datasets consist of web-scraped image collections. This approach, however, makes it very challenging to obtain desirable data such as multiple views of the same object, 3D geometric information, or camera parameters for a large-scale image dataset. In this paper, we propose a 3D-scanned multi-view 2D image dataset of fine-grained category instances with accurate camera calibration parameters. We describe our bi-directional, multi-camera and 3D scanning system and the data collection pipeline. Our target objects are relatively small, highly-detailed fine-grained category instances, such as insects. We present this dataset as a contribution to fine-grained visual categorization, 3D representation learning, and for use in other computer vision tasks.

The final version of the FineView dataset is available at: *https://github.com/byu-vision/fineview*

1. Introduction

Nature and wildlife observation is the practice of noting both the occurrence and abundance of plant or animal species at a specific location and time. Common examples of this type of activity are bird watching (birding), insect collecting, and plant observation (botanizing), and these are widely accepted as both recreational and scientific activities in their respective fields. However, many highly-similar species are difficult to disambiguate; identifying an observed specimen requires expert knowledge and experience in many cases. This hard problem is called Fine-grained Visual Categorization (FGVC) and focuses on differentiating between hard-to-distinguish object classes. Examples of such fine-level classification include discriminating between similar species of plants and animals or identifying the make and model of vehicles, instead of recognizing these objects at a coarse level. An FGVC example of butterflies is shown in Figure 1. These two species have similar colors and shapes, but the patterns on the wings are distinct. When presented with near-identical poses as in the figure, this classification can be performed very effectively by a machine. However, in more extreme conditions of pose, illumination, occlusion, etc, the task becomes much harder. While machines struggle in such scenarios, humans can still find the needed visual cues and differences by factoring in the pose of the butterfly and comparing patterns on common parts; in part, because humans can infer an object's rough 3D shape, understand the lighting and camera angle, and even envision what it would look like from another pose. Humans have developed a 3D understanding of a butterfly because we have seen moving butterflies previously. What if machines had the same information about the object? Information such as object pose, camera angle, object texture, and part labels, would undoubtedly help improve performance on the FGVC task.

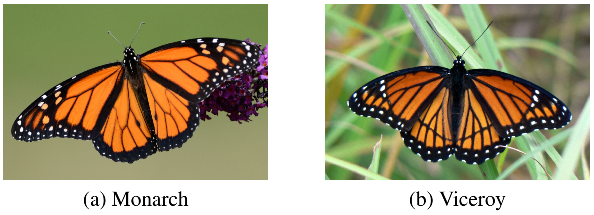

(a) Monarch
(b) Viceroy

Figure 1. **FGVC example of the butterfly**

In recent years, emerging deep learning [27] technologies have made impressive progress in the field of FGVC, as powerful methods for learning feature representations directly from 2D images [29]. Large-scale diverse image collections are essential to training those deep learning models, however, most of these datasets consist primarily of web-scraped images [15, 28, 39, 68] that lack key information such as camera calibration parameters, pose and even an accurate 3D representation. Furthermore, because of large intra-class variations, many state-of-the-art models struggle to disentangle underlying representations, such as 3D and pose information, without explicit supervision. Keypoint, part-segmentation, and correspondence are auxiliary information useful for 3D representation learning, but dense and high-quality annotations are impractical or very limited.

In this paper, we propose the **FineView Dataset**: a 3D scanned multi-view object dataset of fine-grained category instances with accurate camera calibration parameters. We have developed a scanning system that captures multi-view imagery from diverse angles on a viewing sphere, allowing us to obtain a high-resolution multiview 2D image collection with calibrated camera parameters and high-quality 3D point cloud representation. We hope that this work will lay the groundwork for future advances within the field of FGVC and the computer vision research community, generally.

2. Related Work

There are various object-centric 3D scanned CAD datasets [9, 10, 72, 73, 87] that have been proposed for classification, pose-estimation, and 3D representation learning. These datasets don't have fine-grained category classes and they are mostly low-quality and lack fine details on the surface. Other 3D datasets [7,11,13,17,18,42,55,59,62,64,79] have been proposed as realistic 3D objects. Yet other works have proposed synthetic 3D datasets of humans [32,41,53], animals [37, 50, 60, 89, 90] and insects [8, 85], which allow the modulation of 3D object pose and/or generating 2D images from a large variety of camera angles. Nevertheless, the annotations of these datasets are with coarse-level classes, and not fine-grained level categories. There are 3D CAD airplane, chair [30, 33] and bicycle [48] datasets regarded as fine-grained class categories, but these are synthetic CAD data and not real-world data.

The FGVC and computer vision communities have created web-scraped fine-grained 2D image benchmark datasets covering various domains, including vehicles [25, 34, 57, 80, 84]; architectures and buildings [3, 4, 65, 76]; plants [14,40,56,66]; animals [5,6,19,23,43,68,84,88] and insects [47, 66, 69–71, 78]. The main task of those datasets is classification, and each image of those datasets has a fine-grained class label, but they are all single-view images. Furthermore, pose variations between the images within a class are often very limited. Image sets with limited parallax provide limited assistance towards triangulation, and these datasets are thus less desirable for 3D vision applications.

3D reconstruction and novel view synthesis using multi-view single object 2D imagery are some of the most actively investigated topics in the computer vision community. There are various multi-view 2D image datasets [26, 54, 61, 63, 67, 74, 77, 81–83] that have been proposed with synthetic 3D models or with 2D and 3D annotation. In addition, single object sequential Videos can also be regarded as another form of multi-view 2D images [1, 2, 21, 35, 44, 75]. However, object variation in these datasets is scarce and none of them have fine-grained category classes. Fine-grained action recognition is another space for video datasets [31, 38, 49, 52]. Those created for action recognition do not apply to our application of fine-grained object recognition.

There are some laboratory-based systems [55, 67] that can be used to collect multi-view 2D images. Others collect 2D/3D data using a handheld 3D scanner or a handheld video recorder [2, 62, 83], however the camera angle variation of these datasets is somewhat sparse and only from a hemisphere of camera angles. Our target object is relatively small insects like butterflies, and these are usually pinned specimens. A few systems have been proposed for capturing small pinned insects with a similar camera [16, 45, 58] from spherical camera poses, however, those scanning times are more than a few hours per sample and it is hard to collect a large-scale fine-grained object dataset in this manner.

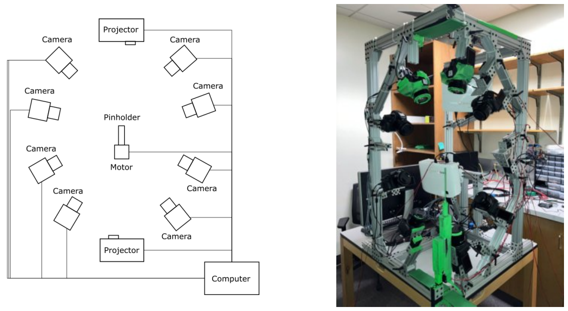
In our work, we propose a system that can capture both multi-view 2D images from a full spherical range of camera angles, and, 3D scanning data that allows us to obtain a high-resolution multi-view 2D image collection with calibrated camera parameters and high-quality 3D point cloud representation. Our data-collecting process is faster than previous methods without implementing industrial-grade products. Our initial dataset includes 360-degree multi-view 2D images from spherical views and 3D point cloud data of fine-grained classes of butterflies and even more data is being actively collected.

3. Dataset Collection System and Process

Our primary recognition targets relatively small insects, such as butterflies. Many web-scraped butterfly images [69] are captured from a top or side angle. These images lack variation in camera viewing angle. Also, a video clip dataset contains some variety of camera angles for the same object [1], but the camera angles are only from the upper hemisphere and low-angle shots are very limited. Many small insects stay on top of leaves, trees, or the ground, and it is difficult to shoot images from underneath angles. Our system is capable of capturing images from many elevation angles approximating a full spherical range and capturing 3D features without using commercial 3D scanners.

3.1. System Overview

The camera array in our system is comprised of 8 DSLR cameras (Panasonic model LUMIX DMC-FZ1000) mounted to aluminum rails as shown in Figure 2b. Two consumer projectors and six softbox lights are used as lighting components, and there is a sample-holding shaft (later called a pinholder) which is mounted on a stepper motor. All of those devices are connected to and controlled by the computer (Raspberry Pi 4B) as shown in Figure 2a.



(a) system schematic diagram (b) Side view of our system

Figure 2. **System overview**

There are two capture modes for the system. The first mode captures 2D multiview high-resolution RGB images illuminated with the softbox lights; the second mode uses structured-light patterns from the projectors to collect 2D RGB images, and both are captured by these 8 cameras. The structured light maps unique markers onto the surface of an object, and 3D coordinates can be calculated by multiview triangulation using the corresponding spots. In order to shoot different images from different camera angles, our system rotates the pinholder using the stepper motor. A hall sensor is attached to the stepper motor and used for positional encoding. There are some advantages of rotating the target sample. It enables us to reduce the number of cameras needed. This is cost-effective and also avoids additional clutter in the background, such as camera mounting rails. Furthermore, we set up green screen backdrops in the background of each camera's view, and this makes it easier to generate the object masks used for downstream processing.

Our target objects are pinned insects and those are placed on the pinholder manually. Focus stacking [12] [24] may improve image quality by using different focus depth images, however, it requires the capture of several images at each angle/viewing location plus extra processing time. We take a balance between data collection time and image quality: we shoot one image for each camera angle with the largest f-number setting in order to capture large focal depth images. We adjust the camera focus for each sample manually and the image-capturing process is automatically performed by a pre-programmed sequence on the controller. The following steps summarize the procedures for collecting data for a single target object:

1). Capture structured-light illuminated 2D RGB images using projectors' illumination.

2). Capture high-resolution 2D RGB images using white light illumination.

3). Rotate the stepper motor which is attached to the pinholder, and repeat these image capturing process. (1) is captured every 180 degrees and (2) is captured every 9 degrees.

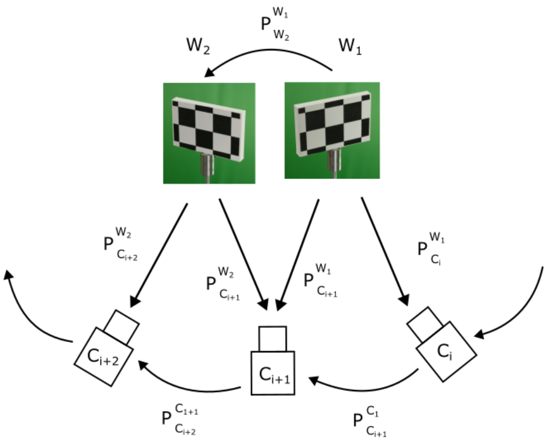


Figure 3. **Relative camera pose of each cameras**

We aim to build a system that captures multi-view 2D images and 3D scans while achieving lower cost, faster capturing time, and fewer manual procedures than an industrial-grade 3D scanning system. We use mostly consumer-grade products to build our system, and the total collecting images is nearly 900 images per sample and it took approximately 7 minutes with a mostly automatic procedure.

3.2. Calibration

Our system requires calibration for successful 3D reconstruction; this calibration includes estimation of both the intrinsic and the extrinsic camera parameters for each of the 8 fixed cameras and also the relative pose of the 8 cameras as the pinholder rotates. First of all, the intrinsic parameters of the 8 cameras are estimated individually using a checkerboard pattern. The 8 cameras reside on two aluminum frame "support arcs" – 4 cameras on each arc – and these two arcs are located 90 degrees from each other relative to the pinholder. The 8 cameras are alternately placed on the two arcs as we ascend in elevation – in the sequence from top to bottom, the even cameras are on one support arc, the odds on the other support arc.

Because of the camera placements and orientations, it is impossible to shoot the same 2D checkerboard image from all 8 cameras at one time. Therefore, we capture checkerboard images with a group of neighboring cameras for the extrinsic parameter estimation, then change checkerboard orientation, and then repeat this process as shown in Figure 3. W is the checkerboard coordinates and P_C^W is the extrinsic parameters/camera pose from coordinate W to camera C, which is calculated by solving the Perspective-n-Point (PnP) problem. Therefore, as the checkerboard orientation

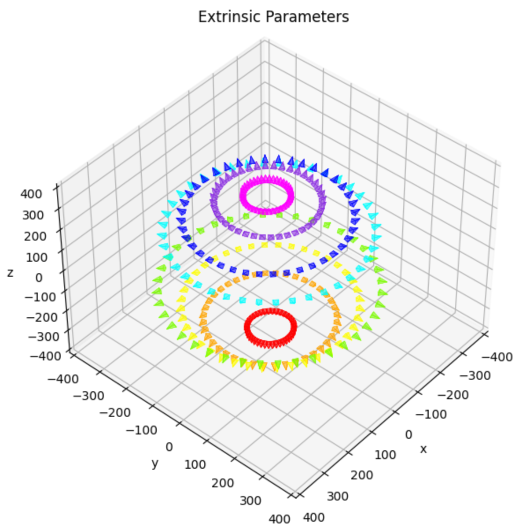
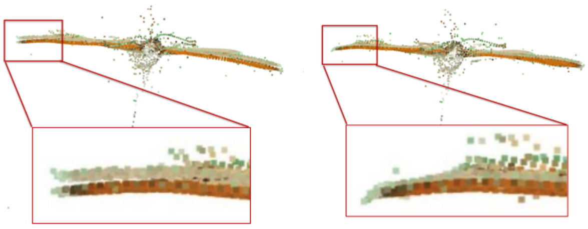


Figure 4. **Camera pose of 8 cameras at each rotated location**

is changed from W_1 to W_2, the relative pose from C_i to C_{i+1} and C_{i+2} can be calculated. We repeat a similar process as we rotate the pinholder and calculate the relative camera pose of the same camera as the checkerboard orientation is changed, then obtain the relative camera pose of all 8 cameras for each angle of the pinholder's rotation.

As described above, camera poses are calculated successively by using the relative poses of neighboring cameras, however, the accumulation error is unfortunately non-negligible. We show an example of 3D reconstructed butterfly object in Figure 5 (See details about 3D reconstruction in following section). These 3D point cloud results are obtained from the front view of a butterfly sample, and you can see a gap between the dorsal and the ventral part of butterfly's wings in figure 5a. To alleviate this problem, we refine each camera's pose using gradient descent optimization. After the optimization process, the same 3d point clouds are shown in 5b, and the gap between the dorsal and the ventral parts is improved.



(a) Before optimization

(b) After optimization

Figure 5. **Comparison of camera pose optimization**

We will elaborate the optimization process here. We use reprojection loss and accumulation loss for this optimization. Detected 2D corner points of the checkerboard are x, and the 3D locations of the corners of the checkerboard are X, the estimated intrinsic parameters are K, and the calculated camera pose/extrinsic parameter is P, then the reprojected 2D corners $x' = KPX$, and the reprojection loss is shown in Eq. 1 where I is the total number of camera position and $d()$ is the L2 norm.

$$L_{repro} = \sum_{i=1}^{I} d(x_i - x'_i)^2 \tag{1}$$

Each camera is located vertically with two neighboring cameras and each camera's pose can be calculated in both the ascending and descending directions along the neighboring camera chain. The estimated camera pose from the two directions is slightly different due to the accumulation of estimation error, and the amount of drift is also non-negligible. The camera pose $P = [R|T]$ where R is the rotation matrix and T is the translation matrix, then P_u and P_d are the estimated pose of the same camera in the up and $down$ directions, respectively. Accumulation loss is shown in Eq. 4 where $trace()$ is the sum of the diagonal elements in a matrix.

$$R_{distance} = \sum_{i=1}^{I} 1 - \cos \frac{trace(R_1 R_2^{\top}) - 1}{2} \tag{2}$$

$$T_{distance} = \sum_{i=1}^{I} d(T_1 - T'_2)^2 \tag{3}$$

$$L_{accum} = \frac{R_{distance} + T_{distance}}{I} \tag{4}$$

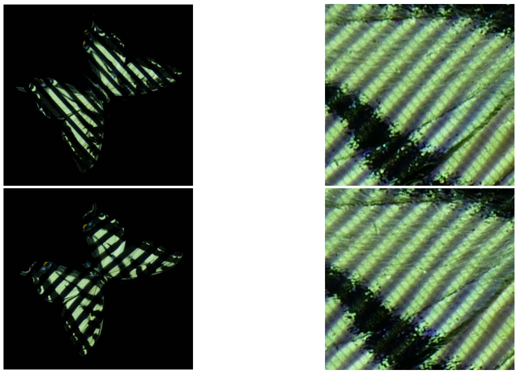
Finally, the total loss is shown in Eq. 5

$$L_{total} = L_{repro} + L_{accum} \tag{5}$$

The extrinsic parameters/camera pose of all 8 cameras at each rotated location are estimated as shown in Figure 4. Each color represents the poses of one of the 8 cameras. The pinholder is rotated each time by 9 degrees, capturing 40 images per camera, for a total of 320 camera pose images for each object specimen/sample. This calibration only has to be done once before collecting data since all the cameras are fixed in the laboratory setting.

3.3. 3D reconstruction

After completing calibration for all of the cameras' intrinsic and extrinsic parameters and the relative camera poses of each rotated position of the pinholder, we capture a set of 320 2D RGB images and two sets of structured-light-illuminated 2D images. To do this, we use projectors to illuminate binary code patterns [46] on the surface of the object as shown in Figure 6. As shown in Figure 6a, we use X-



(a) Binary code pattern X (top) and Y (bottom)

(b) First binary digit image: normal image (top) and bit flipped (bottom)

Figure 6. **Binary code pattern**

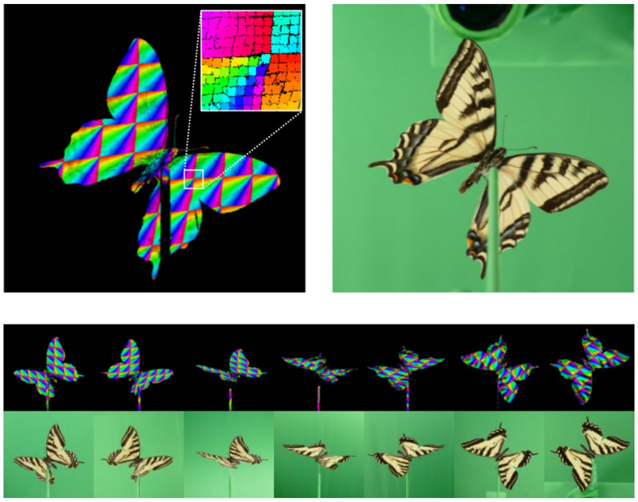
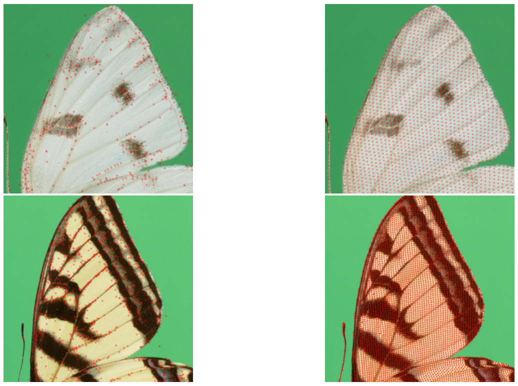


Figure 7. **Unique coded pattern images on the surface of butterfly**



(a) COLMAP feature points

(b) Mapped key points location of our dataset

Figure 8. **Comparison of extracted feature points**

and Y-axis stripe binary code patterns which give us globally unique codes for each point on the surface, facilitating correspondence across the cameras. Figure 6b is the maximum number of patterns that can be projected which is the first binary digit image. This is the narrowest stripe pattern and approximately 1.5 mm per unique binary code, and it is the resolution of the correspondence pattern of our system. The pixel brightness of the stripe area coded as 1 is not consistent due to the color, shape, and material of the surface of the object. The transition edges between 1 and 0 code areas are not always sharp in some regions, and also the dark region of stripe pattern coded as 0 is not completely the low pixel intensity, especially when the image is the narrowest stripe pattern. We use the center of the coded pattern instead of the edge, therefore binary code is preferable to a gray code [22] because the narrowest width of the stripe pattern is twice as wide compared to the binary code. We use the flipped bit pattern of each binary image to determine the 1/0 area by comparing the pixel values of the normal and flipped binary images. This requires twice the number of scanned images but helps to obtain more stable results. Examples of unique coded patterns and RGB images are shown in Figure 7. Each colored square represents a projected unique code location. (For visibility colors are repeated for different unique codes), and the centers of each colored square are used as key points with identical unique codes. we use two projectors to obtain those unique coded patterns on both the upper and lower surfaces of a target object. Here we show the comparison between COLMAP [51] feature extraction and ours in Figure 8. The top images are relatively small butterflies and the bottom are large ones. Both butterfly wings have plain and non-distinctive feature regions, and it is difficult to extract feature points for COLMAP. However, our unique coded pattern is capable of mapping key points in those regions. Our extracted points are more than 5000 per image, whereas COLMAP only detects around 500.

Now we have multi-view images with unique corresponding key points and a 3D point cloud can be calculated by triangulation. A summary of the steps for generating the 3D point cloud representation follows.

1). Process the binary-coded images into a unique-coded image for each camera pose, and extract camera-specific locations for each unique-coded 2D key point as the center of each unique-coded region.

2). Calculate triangulation to obtain the 3D point cloud using the 2D corresponding key points and calibrated camera parameters. This is essentially shooting rays through each camera's respective camera-specific image location and seeing where they meet in 3D space.

3). Conduct outlier filtering of the 3D point cloud. We exclude noisy points which are far above the average distance of nearby points across the cloud.

Dataset	Fine-grained Category	Class	Camara pose	Camera view
BigBIRD [55]	N	100	600	hemisphere
Objectron [2]	N	9	> 10s video	hemisphere
Voynov1 *et al.* [67]	N	107	100	hemisphere
MVImgNet [83]	N	238	10s video	hemisphere
Doan *et al.* [16]	N	13	216	sphere
Ours	**Y**	173	320	**sphere**

Table 1. **Comparison of our dataset to other multi-view 2D dataset.**

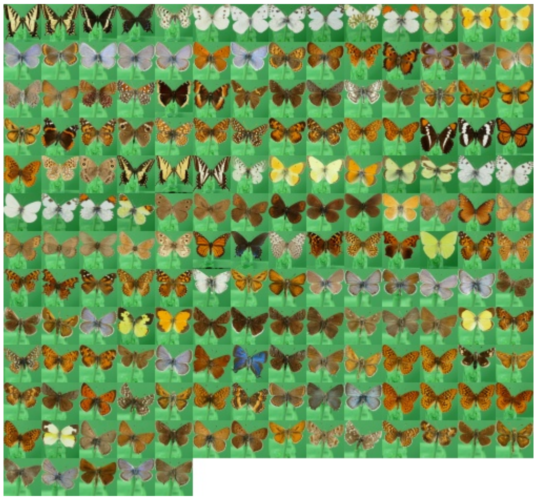


Figure 9. **Butterfly specimens**

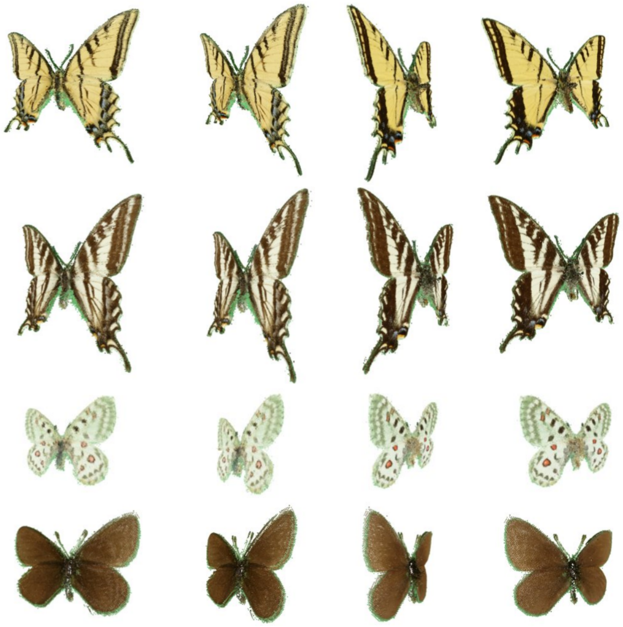


Figure 10. **Reconstructed 3D point clouds**

4. The FineView Dataset

In this section, we describe the details of the dataset, compare it to other datasets, and provide examples of the dataset and its usage. Our dataset contains 68160 images of 173 subcategories belonging to the butterflies category, but we plan to explore more fine-grained categories and build a broader and even larger-scale fine-grained 3D and multi-view dataset in the near future.

4.1. Dataset Details

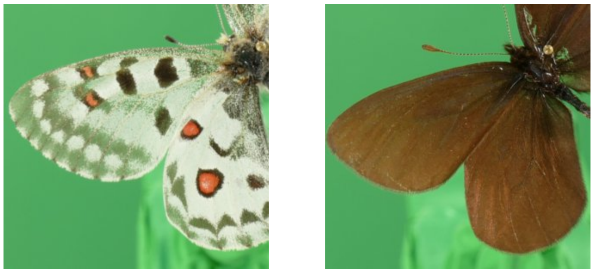
The FineView dataset is composed of 213 butterfly objects (across 173 species) found in North America. Each specimen is dry-preserved and pinned with its wings extended. Each sample's size ranges from 2 cm to 12 cm in diameter. Thumbnails of all species are shown in Figure 9, and there are some butterfly groups with similar shapes and appearances, which is very desirable for FGVC recognition tasks. The data for each object includes multi-view 2D images (320 camera poses) and a 3D point cloud. Each 2D image comes with known extrinsic and intrinsic camera parameters, and the resolution of the multi-view images is between 1.4k × 1.2k (2MB) to 4.2k × 3.6k pixels (18.5MB), and the point cloud size ranges between 6k (0.1MB) to 93k (1.5MB) points. Furthermore, as a byproduct of the multi-view data processing, the 2D object mask for each multi-view image and 2D corresponding key points of each 8 cameras are included for each object.

4.2. Comparison to Other Datasets

We describe the comparison of our dataset to other multi-view 2D datasets in Table 1. Some existing datasets feature large numbers of classes and/or objects, but none of them have fine-grained category classes. To our knowledge, the proposed FineView dataset is the first multi-view 2D image and 3D point cloud dataset with fine-grained category classes. Moreover, our system captures images from a spherical 360-degree view, even from the bottom of the object, and that is a great advantage, especially for small target objects, such as insects.

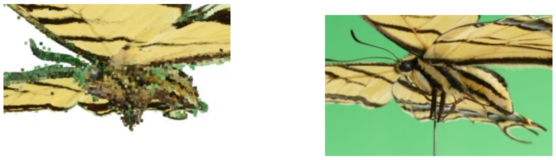
4.3. Dataset Examples

A few examples of reconstructed 3D point clouds are shown in Figure 10. The left two columns are on the dorsal (top) side and the right two columns are on the ventral side (underside). The first and second rows are *Papilio multicaudata* and *Papilio eurymedon*, respectively, some of



(a) Parnassius smintheus (b) Erebia magdalena

Figure 11. **2D images examples**



(a) 3D pointcloud (b) 2D image

Figure 12. **Limitation of 3D reconstruction of our system**

the larger butterflies in our dataset. A butterfly's wings are not completely flat but slightly curved, and both the shape and finely patterned texture are successfully captured. The third row is *Parnassius smintheus*, and its wings are partially translucent as shown in Figure 11a. The fourth row shows *Erebia magdalena*. This species doesn't have a high-contrast pattern, and its wings have less distinctive features than the others as shown in Figure 11b. These species are also successfully reconstructed. However, there are some noisy points, especially at the object boundary that can be observed, and the color of those points sometimes carries a bit of a green background. This is because the pixel colors are alpha-blended between the object and the background in the 2D images at the edge of the object. Also, due to the fact that our system's 3D scan resolution is larger than some of the small and fine structures of the butterfly, we can not perfectly reconstruct some of the finest elements of the body, such as the antennae or legs (see Figure 12).

4.4. Usage of Dataset

Our FineView dataset provides a large variety of camera views of objects from a fine-grained category, butterflies, which gives us rich intra-class variation for the FGVC task. Also, our dataset has known camera poses, corresponding 2D keypoints, object segmentation masks, and a 3D point cloud model, all of which are very labor-intensive to acquire via a human annotator. Those are provided by our data generation process without any manual annotation.

Another line of usage for our dataset is in 3D vision tasks, such as 3D reconstruction. NERF [36] is a novel method for generating views of complex 3D scenes, and moreover, it is one of the hottest topics in 3D computer vision right now. Most web-scraped images or videos don't have camera pose information which is required for training NERF models. COLMAP [51] has become the de-facto standard for structure from motion reconstruction and can be used to estimate camera parameters, but it is difficult to capture the needed keypoints when the object and images of it have limited or non-distinctive features or low-contrast patterns. Typically, it is impossible to estimate the intrinsic parameters for web-scraped images. The FineView dataset has extrinsic and intrinsic camera parameters and a wide variety of camera pose images which makes it a great resource for this 3D vision task as well.

4.5. Butterfly FGVC task example

One of our hypotheses is training deep learning models with various multi-view images of target objects may improve FGVC tasks. Therefore, we created two butterfly FGVC datasets, a Flickr butterfly dataset, and an iNaturalist butterfly dataset. Both are web-scraped image collections of butterflies with Fine-grained species labels. The Flickr butterfly dataset has 150 butterfly species classes of 12k images, which are object-centric single butterflies images with manual segmentation and bounding box annotation. The iNaturalist butterfly dataset has more than 1k species classes and 800K images, which are curated by automatic procedure, and contain multiple objects and non-object-centric images and this is regarded as a challenging dataset compared to the Flickr dataset. We also use the iNaturalist 17 dataset [66] for the existing dataset comparison. Our FineView dataset has 66/37/130 common species classes for the flicker/iNat17/iNat dataset. We train the ImageNet pretrained ResNet-50 model [20] using our FineView dataset. However, catastrophic forgetting issues [86] are well-known when training two distinctive datasets, and normal training on only new data may cause easily overfitting and forgetting problems. We follow the manner of [83] and mix the Flickr/iNat images with our FineView dataset with different selections and ratios. We select FineView images randomly (+random), one consecutive pose image (+consecutive), and a well-distributed pose image (+pose). We use a color jitter except for HUE modification, random resized crop, and with horizontal flip as image augmentation. The accuracy results are shown in Table 2.

Dataset	**iNat**	**iNat17**	**Flicker**
No Fineview	67.97 %	76.51 %	86.47
+random	68.40 %	77.08 %	86.93
+consecutive	68.50 %	76.80 %	87.46
+pose	67.98 %	77.36 %	86.97

Table 2. **Comparison of FGVC task with differently mixed datasets.**

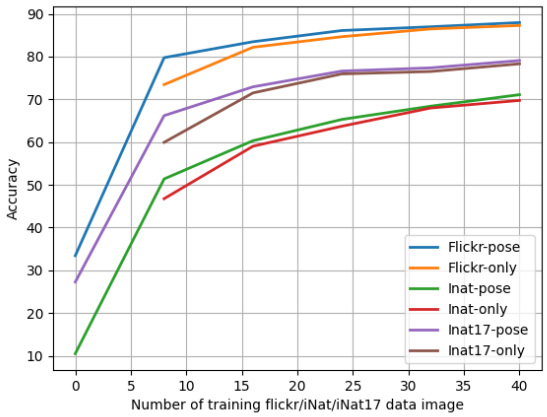


Figure 13. **Comparison between different data ratio**

Training image	PSNR	SSIM	LPIPS
MVImgNet [83]	24.67	0.736	0.310
Fineview	28.78	0.896	0.213

Table 3. **Nerf quantitative comparison with MVImgNet [83].**

As you can see, additional FineView image datasets are better than Flickr/iNat17/iNat-only image-trained models. Let's see how the data ratio affects this performance, we fixed the mixing number of FineView images and changed the number of iNat/Flickr images. The result is shown in Figure 13, and the additional FineView image datasets always improve the accuracy of the image-trained model, which is remarkable when Flickr/iNat17/iNat images are fewer. This simple experiment indicates that the FineView dataset supplies the variation of camera pose and improves model accuracy, especially when training data is scarce even if the Fineview dataset lacks object pose variation.

4.6. NeRF model example

One of the advantages of the FineView dataset is the pre-calibrated extrinsic and intrinsic parameters for real-world spherically captured 360 scene images, which are usually only available for synthetic images. We train vanilla NeRF [36] using the FineVIew dataset of 1/8 scaled images and show the result of a few different synthetic unseen view images in Figure 14. Table 3 shows the Nerf quantitative comparison with MVImgNet [83]. Our Nerf results are better than MVImgNet Nerf results of PSNR, SSIM and LPIPS. The butterfly objects are successfully learned in the NeRF model by using the pre-calibrated extrinsic and intrinsic parameters.

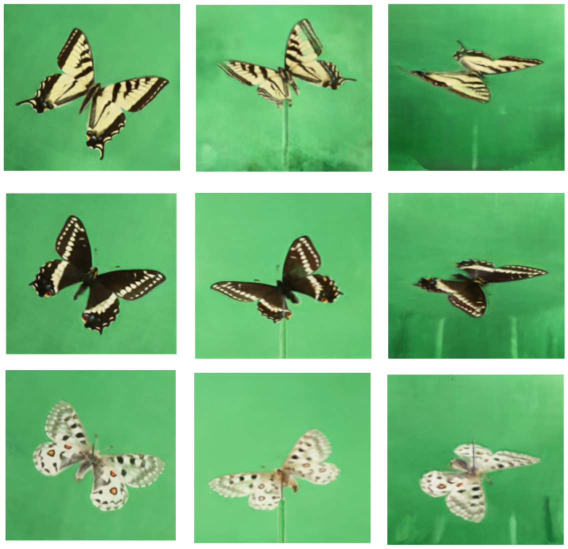


Figure 14. **Different view of NeRF generated image**

5. Conclusion

In this paper, we present the **FineView Dataset**: a 3D scanned multi-view object dataset of fine-grained category instances with accurate camera calibration parameters. We have developed a scanning system that captures multi-view imagery from diverse angles on the viewing sphere, allowing us to obtain a high-resolution multiview 2D image collection with calibrated camera parameters and a high-quality 3D point cloud representation for each object.

The current dataset is limited to butterflies, however, the platform that we have painstakingly built allows additional dataset domains to quickly and easily be captured (just 7 minutes per object). One additional limitation that future work can hopefully overcome is the difficulty with precisely capturing dark-colored fine structures such as legs and antennae.

We envision the FineView dataset being used for diverse real-world applications including autonomous field biology, conservation AI, population monitoring, new species discovery, citizen scientist educational platforms, and, augmented reality. We hope that the dataset will be used by many FGVC and 3D vision researchers and that this will lay the groundwork for future advances in computer vision research.

Acknowledgments This work was supported by the National Science Foundation under Grant No. IIS1651832. We gratefully acknowledge the support of NVIDIA for their GPU donation used in this research. We would like to thank Dr. Alan Myrup and BYU's Bean Life Science Museum for providing both the physical butterfly specimens used in the dataset and identification/taxonomic expertise.

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016. 2

[2] Adel Ahmadyan, Liangkai Zhang, Artsiom Ablavatski, Jianing Wei, and Matthias Grundmann. Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7822–7831, 2021. 2, 6

[3] Connor Anderson, Adam Teuscher, Elizabeth Anderson, Alysia Larsen, Josh Shirley, and Ryan Farrell. Have fun storming the castle(s)! In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3703–3712, January 2021. 2

[4] Björn Barz and Joachim Denzler. Wikichurches: A fine-grained dataset of architectural styles with real-world challenges. *arXiv preprint arXiv:2108.06959*, 2021. 2

[5] Thomas Berg, Jiongxin Liu, Seung Woo Lee, Michelle L. Alexander, David W. Jacobs, and Peter N. Belhumeur. Birdsnap: Large-scale fine-grained visual categorization of birds. In *Proc. Conf. Computer Vision and Pattern Recognition (CVPR)*, June 2014. 2

[6] Thomas Berg, Jiongxin Liu, Seung Woo Lee, Michelle L Alexander, David W Jacobs, and Peter N Belhumeur. Birdsnap: Large-scale fine-grained visual categorization of birds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2011–2018, 2014. 2

[7] Anders Bjorholm Dahl, Patrick Møller Jensen, Carsten Gundlach, Rebecca Engberg, Hans Martin Kjer, and Vedrana Andersen Dahl. Bugnist–a new large scale volumetric 3d image dataset for classification and detection. *arXiv e-prints*, pages arXiv–2304, 2023. 2

[8] Kuo Cai, Jiangtao Li, Yudong Wang, Andi Lan, and Huiling Zhou. A method of establishing a synthetic dataset for stored-grain insects. In *2021 7th IEEE International Conference on Network Intelligence and Digital Content (IC-NIDC)*, pages 153–157, 2021. 2

[9] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2

[10] Sungjoon Choi, Qian-Yi Zhou, Stephen Miller, and Vladlen Koltun. A large dataset of object scans. *arXiv:1602.02481*, 2016. 2

[11] Sungjoon Choi, Qian-Yi Zhou, Stephen Miller, and Vladlen Koltun. A large dataset of object scans. *arXiv preprint arXiv:1602.02481*, 2016. 2

[12] Paolo Clini, Nicoletta Frapiccini, Maura Mengoni, Roberto Nespeca, and Laura Ruggeri. Sfm technique and focus stacking for digital documentation of archaeological artifacts. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 41:229–236, 2016. 3

[13] Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F Yago Vicente, Thomas Dideriksen, Himanshu Arora, Matthieu Guillaumin, and Jitendra Malik. Abo: Dataset and benchmarks for real-world 3d object understanding. *CVPR*, 2022. 2

[14] Riccardo de Lutio, John Y Park, Kimberly A Watson, Stefano D'Aronco, Jan D Wegner, Jan J Wieringa, Melissa Tulig, Richard L Pyle, Timothy J Gallaher, Gillian Brown, et al. The herbarium 2021 half–earth challenge dataset and machine learning competition. *Frontiers in Plant Science*, 12:3320, 2022. 2

[15] Jia Deng, Wei Dong, R. Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009. 1

[16] Thanh-Nghi Doan and Chuong V Nguyen. A low-cost digital 3d insect scanner. *Information Processing in Agriculture*, 2023. 2, 6

[17] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2553–2560. IEEE, 2022. 2

[18] Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3d-future: 3d furniture shape with texture. *International Journal of Computer Vision*, 129:3313–3337, 2021. 2

[19] Tejas Gokhale, Shailaja Sampat, Zhiyuan Fang, Yezhou Yang, and Chitta Baral. Blocksworld revisited: Learning and reasoning to generate event-sequences from image pairs, 2019. 2

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

[21] Philipp Henzler, Jeremy Reizenstein, Patrick Labatut, Roman Shapovalov, Tobias Ritschel, Andrea Vedaldi, and David Novotny. Unsupervised learning of 3d object categories from videos in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4700–4709, 2021. 2

[22] Seiji Inokuchi. Range imaging system for 3-d object recognition. *ICPR, 1984*, 1984. 5

[23] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. Novel dataset for fine-grained image categorization. In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 2011. 2

[24] G Kontogianni, R Chliverou, A Koutsoudis, G Pavlidis, and A Georgopoulos. Enhancing close-up image based 3d digitisation with focus stacking. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42:421–425, 2017. 3

[25] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *2013 IEEE International Conference on Computer Vision Workshops*, pages 554–561, 2013. 2

[26] Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *2011 IEEE international conference on robotics and automation*, pages 1817–1824. IEEE, 2011. 2

[27] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. 521(7553):436–444. Bandiera_abtest: a Cg_type: Nature Research Journals Number: 7553 Primary_atype: Reviews Publisher: Nature Publishing Group Subject_term: Computer science;Mathematics and computing Subject_term_id: computer-science;mathematics-and-computing. 1

[28] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common Objects in Context. In *ECCV*, 2014. 1

[29] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear CNNs for fine-grained visual recognition. 1

[30] Xinhai Liu, Zhizhong Han, Yu-Shen Liu, and Matthias Zwicker. Fine-grained 3d shape classification with hierarchical part-view attentions. *IEEE Transactions on Image Processing*, 2021. 2

[31] Yi Liu, Limin Wang, Yali Wang, Xiao Ma, and Yu Qiao. Fineaction: A fine-grained video dataset for temporal action localization. *IEEE Transactions on Image Processing*, 31:6937–6950, 2022. 2

[32] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: a skinned multi-person linear model. 34(6):1–16. 2

[33] Ling Luo, Yulia Gryaditskaya, Yongxin Yang, Tao Xiang, and Yi-Zhe Song. Fine-grained vr sketching: Dataset and insights. In *2021 International Conference on 3D Vision (3DV)*, pages 1003–1013. IEEE, 2021. 2

[34] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 2

[35] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 2019. 2

[36] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 7, 8

[37] Jiteng Mu, Weichao Qiu, Gregory D. Hager, and Alan L. Yuille. Learning from synthetic animals. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[38] Jonathan Munro and Dima Damen. Multi-modal Domain Adaptation for Fine-grained Action Recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[39] Maria-Elena Nilsback and Andrew Zisserman. Automated Flower Classification over a Large Number of Classes. In *Sixth Indian Conference on Computer Vision, Graphics & Image Processing (ICCVGIP)*, 2008. 1

[40] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*, Dec 2008. 2

[41] Ahmed A. A. Osman, Timo Bolkart, and Michael J. Black. STAR: Sparse trained articulated human body regressor. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, volume 12351, pages 598–613. Springer International Publishing. Series Title: Lecture Notes in Computer Science. 2

[42] Keunhong Park, Konstantinos Rematas, Ali Farhadi, and Steven M. Seitz. Photoshape: Photorealistic materials for large-scale shape collections. *ACM Trans. Graph.*, 37(6), Nov. 2018. 2

[43] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. The oxford-iiit pet dataset. 2

[44] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 2

[45] Fabian Plum and David Labonte. scant—an open-source platform for the creation of 3d models of arthropods (and other small objects). *PeerJ*, 9:e11155, 2021. 2

[46] Jeffrey L Posdamer and Martin D Altschuler. Surface measurement by space-encoded projected beam systems. *Computer graphics and image processing*, 18(1):1–17, 1982. 4

[47] project. Butterfly dataset. https://universe.roboflow.com/project-9zfbb/butterfly-lzhkf, jun 2023. visited on 2023-08-23. 2

[48] Lyle Regenwetter, Brent Curry, and Faez Ahmed. Biked: A dataset for computational bicycle design with machine learning benchmarks. *Journal of Mechanical Design*, 144(3):031706, 2022. 2

[49] Marcus Rohrbach, Anna Rohrbach, Michaela Regneri, Sikandar Amin, Mykhaylo Andriluka, Manfred Pinkal, and Bernt Schiele. Recognizing fine-grained and composite activities using hand-centric features and script data. *International Journal of Computer Vision*, pages 1–28, 2015. 2

[50] Artsiom Sanakoyeu, Vasil Khalidov, Maureen S. McCarthy, Andrea Vedaldi, and Natalia Neverova. Transferring dense pose to proximal animal classes. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5232–5241. IEEE. 2

[51] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 5, 7

[52] Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. Finegym: A hierarchical video dataset for fine-grained action understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[53] Roman Shapovalov, David Novotny, Benjamin Graham, Patrick Labatut, and Andrea Vedaldi. DensePose 3d: Lifting canonical surface maps of articulated objects to the third dimension. version: 1. 2

[54] Rakesh Shrestha, Siqi Hu, Minghao Gou, Ziyuan Liu, and Ping Tan. A real world dataset for multi-view 3d reconstruction. In *European Conference on Computer Vision*, pages 56–73. Springer, 2022. 2

[55] Arjun Singh, James Sha, Karthik S Narayan, Tudor Achim, and Pieter Abbeel. Bigbird: A large-scale 3d database of object instances. In *2014 IEEE international conference on robotics and automation (ICRA)*, pages 509–516. IEEE, 2014. 2, 6

[56] Daniel Steininger, Andreas Trondl, Gerardus Croonen, Julia Simon, and Verena Widhalm. The cropandweed dataset: A multi-modal learning approach for efficient crop and weed manipulation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3729–3738, January 2023. 2

[57] Daniel Steininger, Verena Widhalm, Julia Simon, Andreas Kriegler, and Christoph Sulzbachner. The aircraft context dataset: Understanding and optimizing data variability in aerial domains. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 3823–3832, October 2021. 2

[58] Bernhard Ströbel, Sebastian Schmelzle, Nico Blüthgen, and Michael Heethoff. An automated device for the digitization and 3d modelling of insects, combining extended-depth-of-field and all-side multi-view imaging. *ZooKeys*, (759):1, 2018. 2

[59] Yongzhi Su, Mingxin Liu, Jason Rambach, Antonia Pehrson, Anton Berg, and Didier Stricker. Ikea object state dataset: A 6dof object pose estimation dataset and benchmark for multi-state assembly objects, 2021. 2

[60] Robert W Sumner and Jovan Popović. Deformation transfer for triangle meshes. *ACM Transactions on graphics (TOG)*, 23(3):399–405, 2004. 2

[61] Xingyuan Sun, Jiajun Wu, Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Tianfan Xue, Joshua B Tenenbaum, and William T Freeman. Pix3d: Dataset and methods for single-image 3d shape modeling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[62] Xiao Fu Yuxin Wang Jiawei Ren Liang Pan Wayne Wu Lei Yang Jiaqi Wang Chen Qian Dahua Lin Ziwei Liu Tong Wu, Jiarui Zhang. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[63] Jonathan Tremblay, Moustafa Meshry, Alex Evans, Jan Kautz, Alexander Keller, Sameh Khamis, Thomas Müller, Charles Loop, Nathan Morrical, Koki Nagano, et al. Rtmv: A ray-traced multi-view synthetic dataset for novel view synthesis. *arXiv preprint arXiv:2205.07058*, 2022. 2

[64] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Duc Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *International Conference on Computer Vision (ICCV)*, 2019. 2

[65] Grant Van Horn, Steve Branson, Ryan Farrell, Scott Haber, Jessie Barry, Panos Ipeirotis, Pietro Perona, and Serge Belongie. Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 595–604, 2015. 2

[66] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8769–8778, 2018. 2, 7

[67] Oleg Voynov, Gleb Bobrovskikh, Pavel Karpyshev, Saveliy Galochkin, Andrei-Timotei Ardelean, Arseniy Bozhenko, Ekaterina Karmanova, Pavel Kopanev, Yaroslav Labutin-Rymsho, Ruslan Rakhimov, et al. Multi-sensor large-scale dataset for multi-view 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21392–21403, 2023. 2, 6

[68] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical report, 2011. 1, 2

[69] Josiah Wang, Katja Markert, and Mark Everingham. Learning models for object recognition from natural language descriptions. In *Proceedings of the British Machine Vision Conference*, 2009. 2

[70] Xiaoping Wu, Chi Zhan, Yu-Kun Lai, Ming-Ming Cheng, and Jufeng Yang. Ip102: A large-scale benchmark dataset for insect pest recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[71] Xiaoping Wu, Chi Zhan, Yu-Kun Lai, Ming-Ming Cheng, and Jufeng Yang. Ip102: A large-scale benchmark dataset for insect pest recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8787–8796, 2019. 2

[72] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015. 2

[73] Yu Xiang, Roozbeh Mottaghi, and Silvio Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2014. 2

[74] Yu Xiang, Roozbeh Mottaghi, and Silvio Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *IEEE winter conference on applications of computer vision*, pages 75–82. IEEE, 2014. 2

[75] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017. 2

[76] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010. 2

[77] Haozhe Xie, Hongxun Yao, Shengping Zhang, Shangchen Zhou, and Wenxiu Sun. Pix2vox++: Multi-scale context-

aware 3d object reconstruction from single and multiple images. *International Journal of Computer Vision*, 128(12):2919–2935, 2020. 2

[78] Juanying Xie, Qi Hou, Yinghuan Shi, Lv Peng, Liping Jing, Fuzhen Zhuang, Junping Zhang, Xiaoyang Tang, and Shengquan Xu. The automatic identification of butterfly species. *arXiv preprint arXiv:1803.06626*, 2018. 2

[79] Mutian Xu, Pei Chen, Haolin Liu, and Xiaoguang Han. To-scene: A large-scale dataset for understanding 3d tabletop scenes. In *ECCV*, 2022. 2

[80] Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. A large-scale car dataset for fine-grained categorization and verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3973–3981, 2015. 2

[81] Zhenpei Yang, Zaiwei Zhang, and Qixing Huang. Hm3d-abo: A photo-realistic dataset for object-centric multi-view 3d reconstruction. *arXiv preprint arXiv:2206.12356*, 2022. 2

[82] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1790–1799, 2020. 2

[83] Xianggang Yu, Mutian Xu, Yidan Zhang, Haolin Liu, Chongjie Ye, Yushuang Wu, Zizheng Yan, Chenming Zhu, Zhangyang Xiong, Tianyou Liang, et al. Mvimgnet: A large-scale dataset of multi-view images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9150–9161, 2023. 2, 6, 7, 8

[84] Xiu-Shen Wei Yongshun Zhang Fumin Shen Jianxin Wu Jian Zhang Heng Tao Shen Zeren Sun, Yazhou Yao. Webly supervised fine-grained recognition: Benchmark datasets and an approach. In *IEEE International Conference on Computer Vision (ICCV)*, 2021. 2

[85] Xiaozheng Zhang, Yongsheng Gao, and Terry Caelli. Primitive-based 3d structure inference from a single 2d image for insect modeling: Towards an electronic field guide for insect identification. In *2010 11th International Conference on Control Automation Robotics & Vision*, pages 866–871. IEEE, 2010. 2

[86] Yang Zhang, Fuli Feng, Chenxu Wang, Xiangnan He, Meng Wang, Yan Li, and Yongdong Zhang. How to retrain recommender system? a sequential meta-learning method. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1479–1488, 2020. 7

[87] Qingnan Zhou and Alec Jacobson. Thingi10k: A dataset of 10,000 3d-printing models. *arXiv preprint arXiv:1605.04797*, 2016. 2

[88] Ding-Nan Zou, Song-Hai Zhang, Tai-Jiang Mu, and Min Zhang. A new dataset of dog breed images and a benchmark for fine-grained classification. *Computational Visual Media*, 2020. 2

[89] Silvia Zuffi, Angjoo Kanazawa, Tanya Berger-Wolf, and Michael J. Black. Three-d safari: Learning to estimate zebra pose, shape, and texture from images ”in the wild”. 2

[90] Silvia Zuffi, Angjoo Kanazawa, and Michael J. Black. Lions and tigers and bears: Capturing non-rigid, 3d, articulated shape from images. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3955–3963. IEEE. 2

Supplementary Material

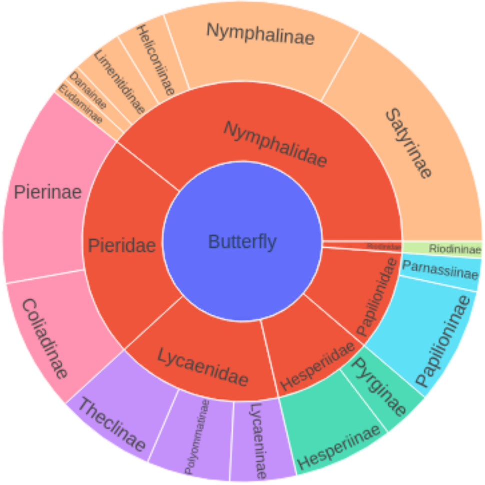


Figure 15. **Butterfly family taxonomy of Fineview dataset**.

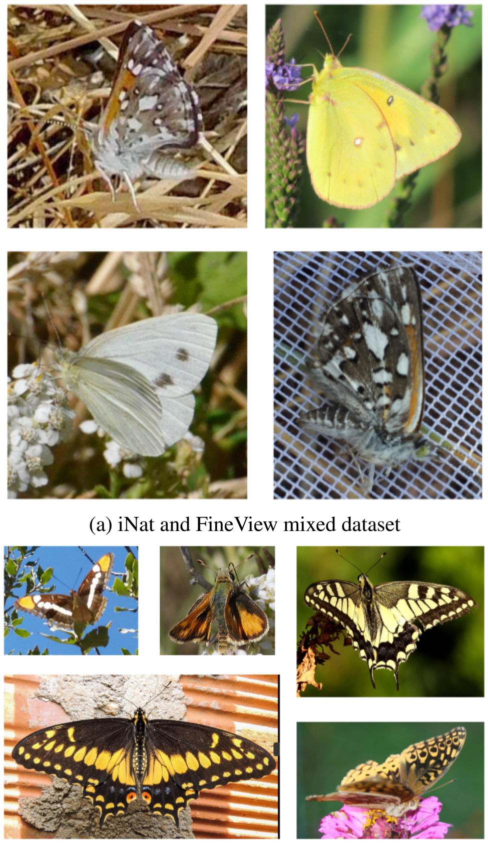
A. FineView dataset taxonomy

In taxonomy, the family rank in the classification of organisms is between genus and order, which is grouped by their common attributes. Butterflies in the same family have some common features, such as shape and color, and it would be subsidiary information for FGVC task. Figure 15 shows the butterfly family and subfamily taxonomy.

B. Further investigation of FGVC task

We investigate the breakdown of incorrect classification of each trained model. Figure 16 shows examples of miss-classified test images of iNat and Fineview mixed dataset-trained model and iNat-only dataset-trained model. The typical misclassified examples of the iNat-only dataset-trained model are certain butterfly poses that extend their wings. This is similar to the butterfly pose of the FineView dataset. The major misclassified examples by The mixed dataset-trained model are the self-occluded butterfly (only certain sides are visible) and the closed-wing pose butterflies.

These results indicate the mixed dataset-trained model accuracy is better than the iNat-only model for certain object pose cases because adding the Fineview dataset reinforces the variety of pose distribution of the training dataset when we use a simple Resnet classification model, and this supports the hypothesis that the classification accuracy depends on the object pose distribution of the training dataset. Furthermore, the FineView mixed-trained model is better especially when the base training dataset is scarce, these results suggest a well-distributed camera pose of the training dataset is crucial for the FGVC task. The FineView dataset is effective for FGVC tasks although the butterfly of the FineView dataset lacks object pose variation. For future work, the FineView dataset can be applied to the object pose-aware classification models for FGVC tasks, which could potentially improve classification accuracy.



(a) iNat and FineView mixed dataset

(b) iNat-only dataset

Figure 16. **Examples of incorrect classification of each trained model**

C. Additional Nerf model examples

One of the advantages of the FineView dataset is the sphere angle distribution of captured images, which is bi-directional 360-degree camera poses. Figure 17 shows several unseen views of synthetic Nerf model-generated images. This camera pose trajectory is along one direction from top to bottom of the sphere of a butterfly. The left column images (*from top image to bottom*) are from top to front view angle and the right column images are from front to bottom view angle camera poses. A particularly eye-catching result is that the butterfly object is invisible in the front view angle camera pose image (*the right top im-*

age in Figure 17). We show the comparison of unseen views of Nerf-generated images (*even rows*) and ground truth test images (*odd rows*) in Figure 18. Horizontal view images (*center column*) are relatively unclear compared to other view images visually and PSNR, SSIM, and LPIPS are approximately 5% worse than other views. We assume the vanilla Nerf model can not capture the horizontal view of the butterfly's body because the butterfly has thin and flat shapes and the antennas and legs are invisible in all generated images. Those flat shapes and fine structures are challenging not only for Nerf models but also for general 3D reconstruction and 3D modeling tasks, and it is a significant research topic for the computer vision community. This is another potential use case of the FineView dataset.

D. FineView dataset examples

Figure 19 shows several sets of examples of multi-view 2D RGB, mask, and the corresponding images. These images have the same pinholder location but are captured by 8 cameras. The mask images capture small structures of butterflies, such as antennae and wing shape. The corresponding key points are consistent between different views. This auxiliary information is labor-intensive for human annotators, but Our proposed system can automatically capture those images.

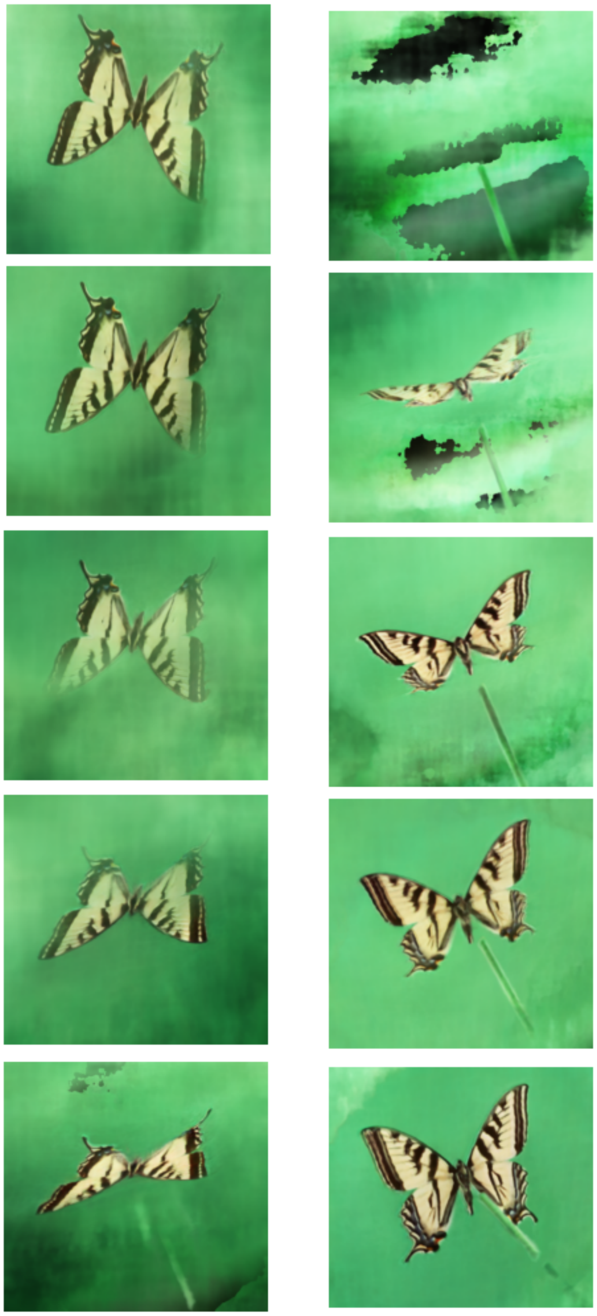


Figure 17. **Various unseen views of Nerf generated images**.

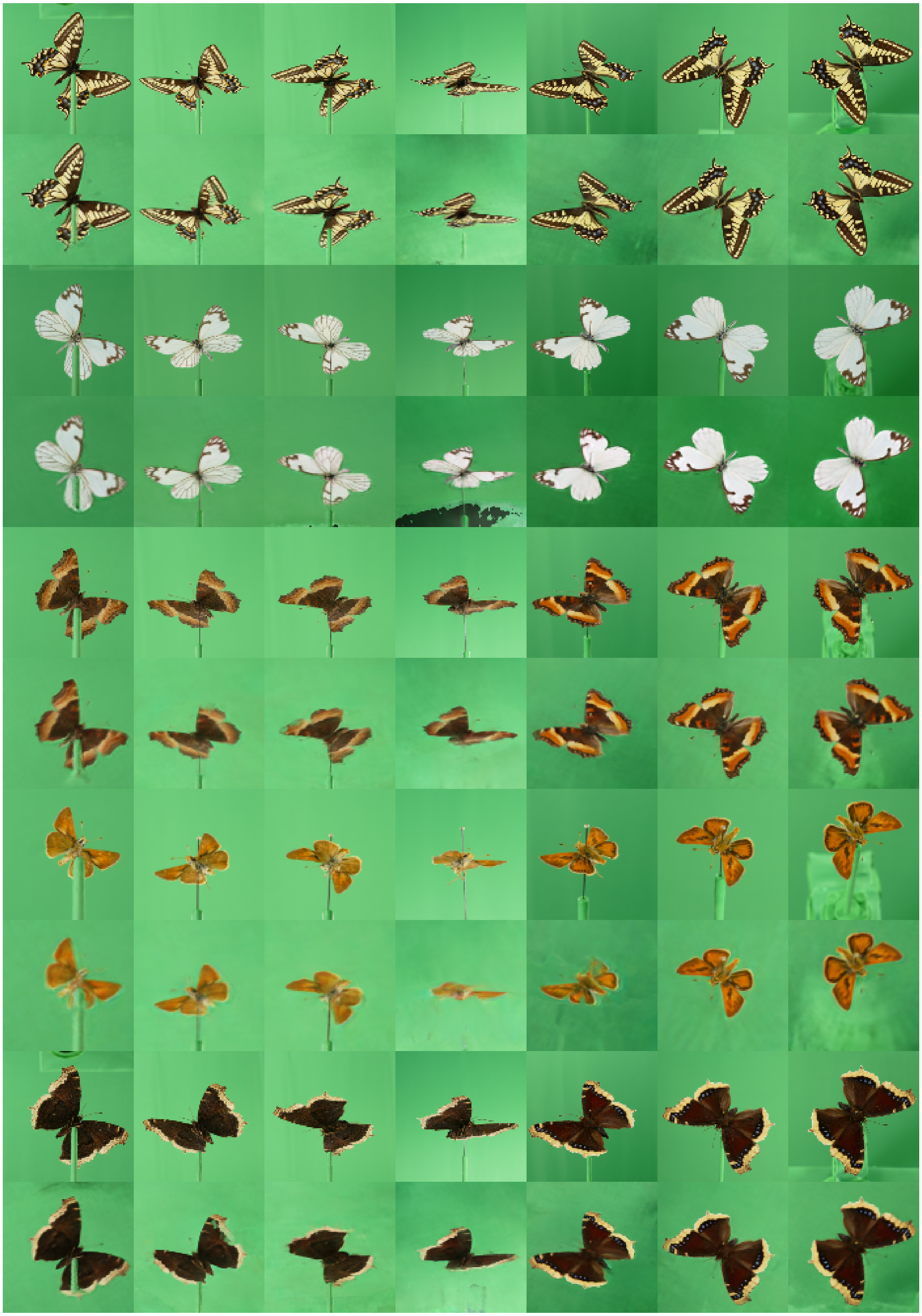


Figure 18. **Ground Truth images (*odd rows*) vs untrained view of Nerf generated images (*even rows*).**

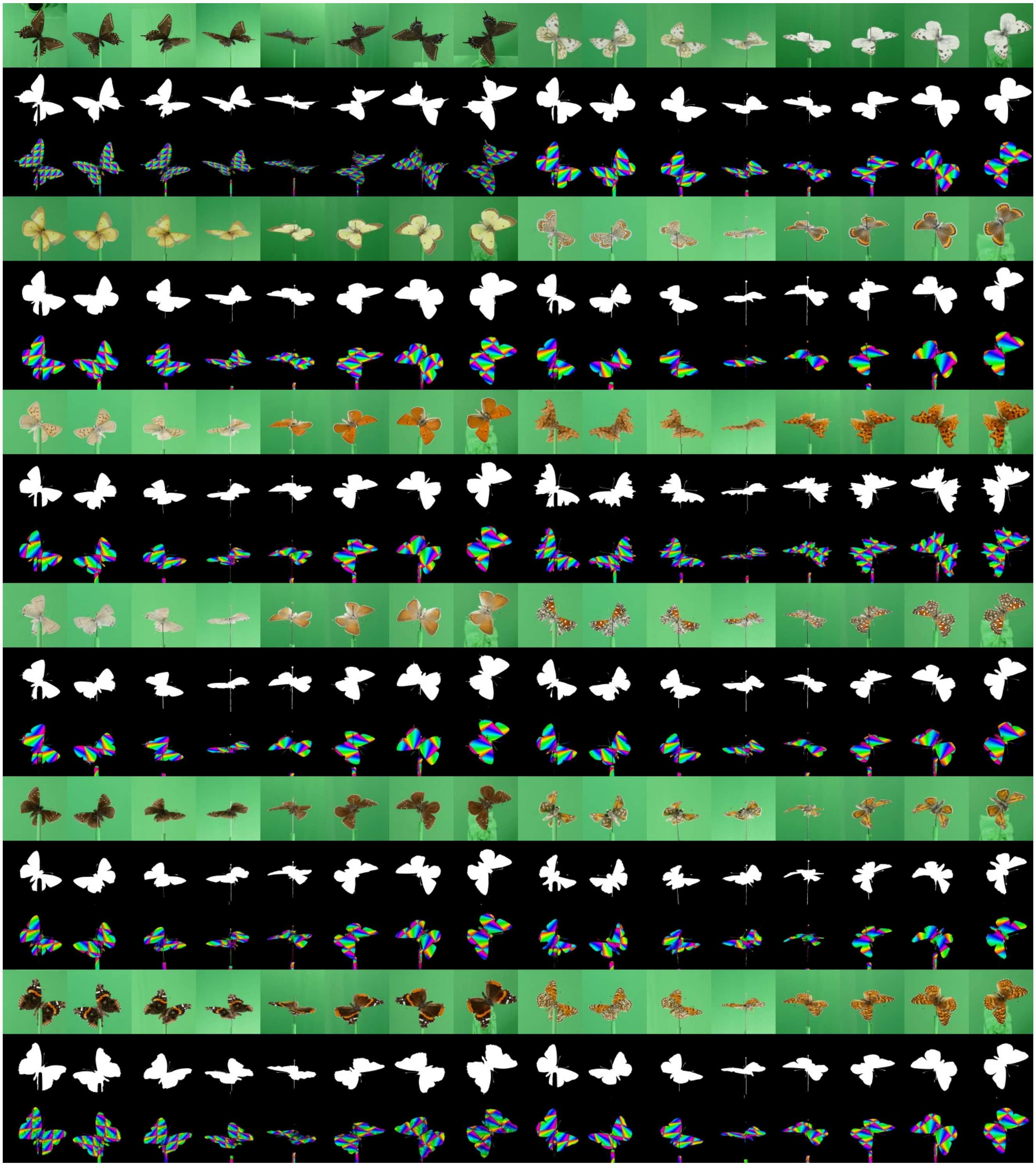


Figure 19. **Examples of multi-view 2D RGB, mask and corresponding images**.