black hat ASIA 2021
May 6-7, 2021
BRIEFINGS

Hiding Objects from Computer Vision by Exploiting Correlation Biases

Yin Minn Pa Pa, Paul Ziegler, Masaki Kamizono

Dr. Yin Minn Pa Pa

- Originally from Myanmar, currently works as a senior researcher and manager at Deloitte Japan
- Research interests include network security, malware analysis, IoT security, web security and AI security
- Conducted several International research collaborations with universities in Myanmar, US, Netherland and Germany
- https://www.yinminnpapa.com

Deloitte.
デロイト トーマツ

Paul S. Ziegler

- Founder of Reflare Ltd, originally from Germany, living and working in Japan
- Consultant and advisor on information security topics to corporate and consultancy customers with a focus on research and development
- Self taught with a comparatively shallow but in turn wide understanding of information security

reFlare

Masaki Kamizono

- CTO and Partner of Deloitte Japan
- Engages in research and new service development
- Manages large-scale projects and designs socially supportive demonstrational experiments that utilize research results
- Contributes to human resource development through the publication of papers, lectures

Executive Summary of Results

- Computer vision systems have an implicit correlation bias that must be accounted for.
- It is possible to completely hide objects from computer vision systems by contextualizing them with other - non-correlating - objects.
- If detection of certain objects through a computer vision system is critical, then training on specialized data that equalizes the training biases can mitigate the risk.

Stop sign hidden by association with food
Major Cloud Vision AIs: 0% / YOLOv3: 0% / Resnet: 0%
COCO #98029

Important Notes

- Unless stated otherwise, all evaluations are performed against YOLOv3 trained on COCO 2017.
- “Major Cloud Vision AIs” refers to computer vision APIs provided by Google and Microsoft. In order not to single one out specifically, we will only represent this category as a total detection percentage among both.
- All evaluations against cloud vision AIs were performed on March 8th 2021. It is likely that future updates to these systems will lead to changes in detection behavior.

Recap of Computer Vision Terminology

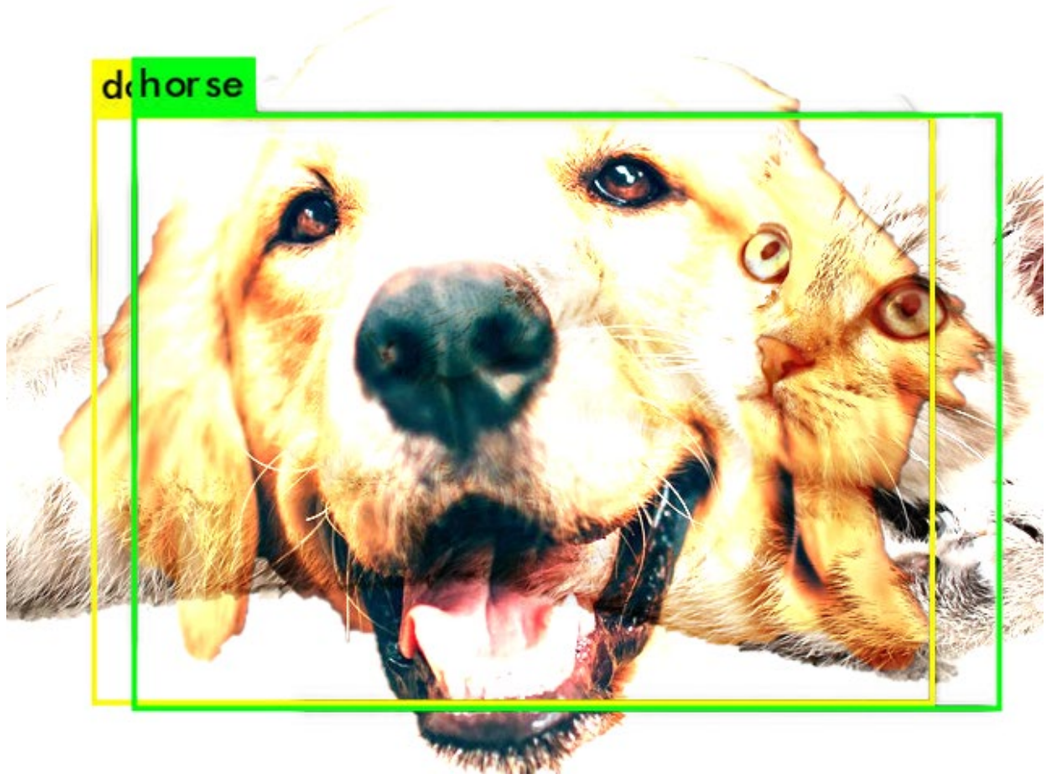
- **Image classification:** There is a stop sign in this picture.
- **Object detection:** There is a stop sign at this location in the picture.
- **Object segmentation:** This is the outline of a stop sign in the picture.
- **Certainty:** I am X% certain that there is a stop sign in this picture.
- **Threshold:** I will detect a stop sign in this picture if my certainty is over X%.

COCO Dataset #34063

Phase 1: Random Image Manipulation

By automatically generating thousands of composite images, certain classes of identification errors can be established.

Misclassification In The Same General Gategory
(e.g. dogs and cats seen as horses)



Caused by the mixing of features used to identify each class.

In most current computer vision systems, detection is done using feature clusters extracted through convolutional layers.

Therefore, combining features from different classes can easily lead to wrong results in the same general category.

Misclassification As Part of Something Else
(e.g. plant on a shirt worn by a person seen as a person)

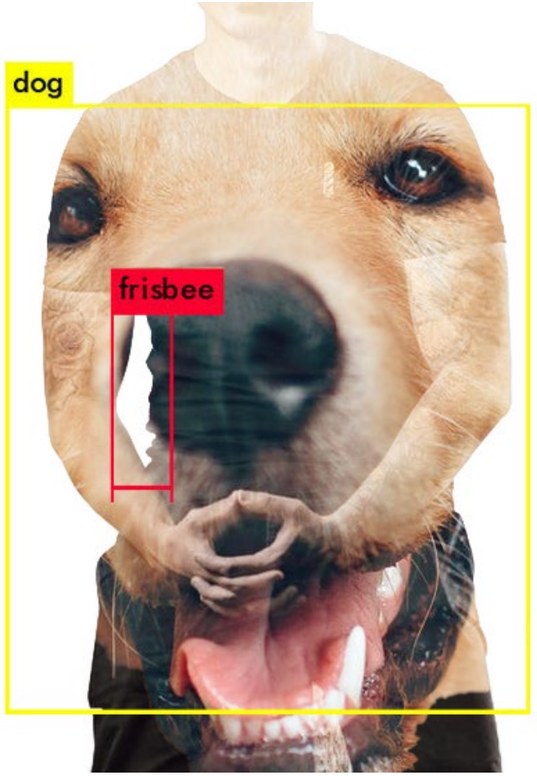
Caused by an unclear logical line between objects.

A person standing in front of a house should clearly be detected as a person, but should a plant on a shirt be detected as a plant?

What about a floral pattern on a round table?

The categorization is difficult for computer vision because it is contextual for humans.

Misclassification By Correlation
(e.g. round object next to dog seen as a frisbee)



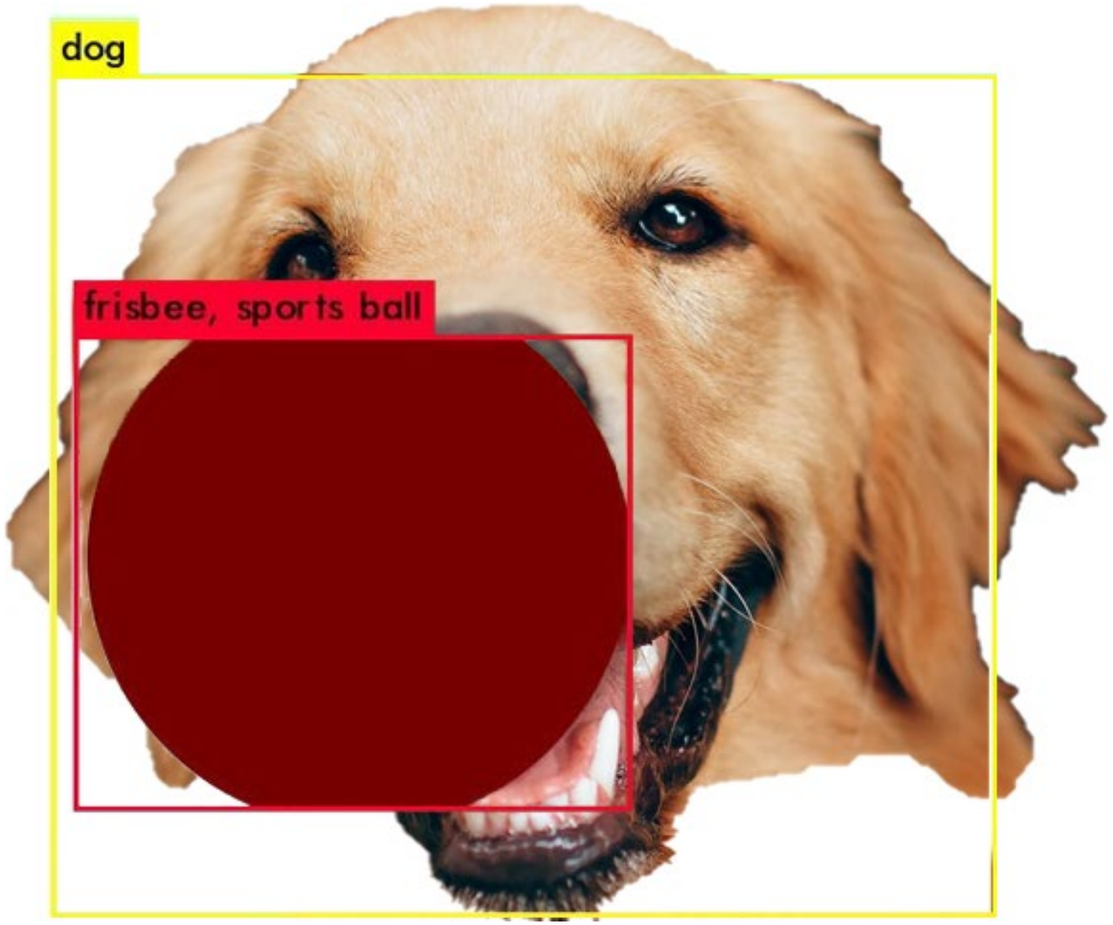
Caused by a high correlation between the two classes in the training dataset.

Since dogs and frisbees appear together in the COCO dataset, a network trained on it will tend to identify one more easily of the other has been detected.

Importantly, the bias **exists in nature** and not just in the dataset. Dogs and frisbees are indeed often seen together.

Examples of Dogs and Frisbees in COCO

We can abuse this correlation bias to have any round shape detected as a frisbee.



Phase 2: Automatic Adversarial Generation

We can establish what classes highly correlate and what classes do not by analyzing public datasets. From there, the process of creating composites that are not detected can be automated.

What correlates with stop signs?

Object	Correlation %
car	160.413515
person	136.106909
traffic light	23.651034
truck	22.541604
bicycle	14.069592
handbag	10.993444
bus	10.136157
motorcycle	9.127584
fire hydrant	6.30358
backpack	6.253152
potted plant	5.648008
umbrella	4.639435
chair	3.983863
bird	3.429148

This table shows how high the correlation to stop signs is in the COCO dataset.

Note that since many pictures containing stop signs include several cars or people as well, the correlation for these classes exceeds 100%.

Out of 90 categories available, only 11 have a significant overlap (>5%) with stop signs. The others can be said to rarely appear in the same image.

Automatic Generation Process

Step 1: Select a random picture with the target class, make sure it's detected.

COCO Dataset #46526

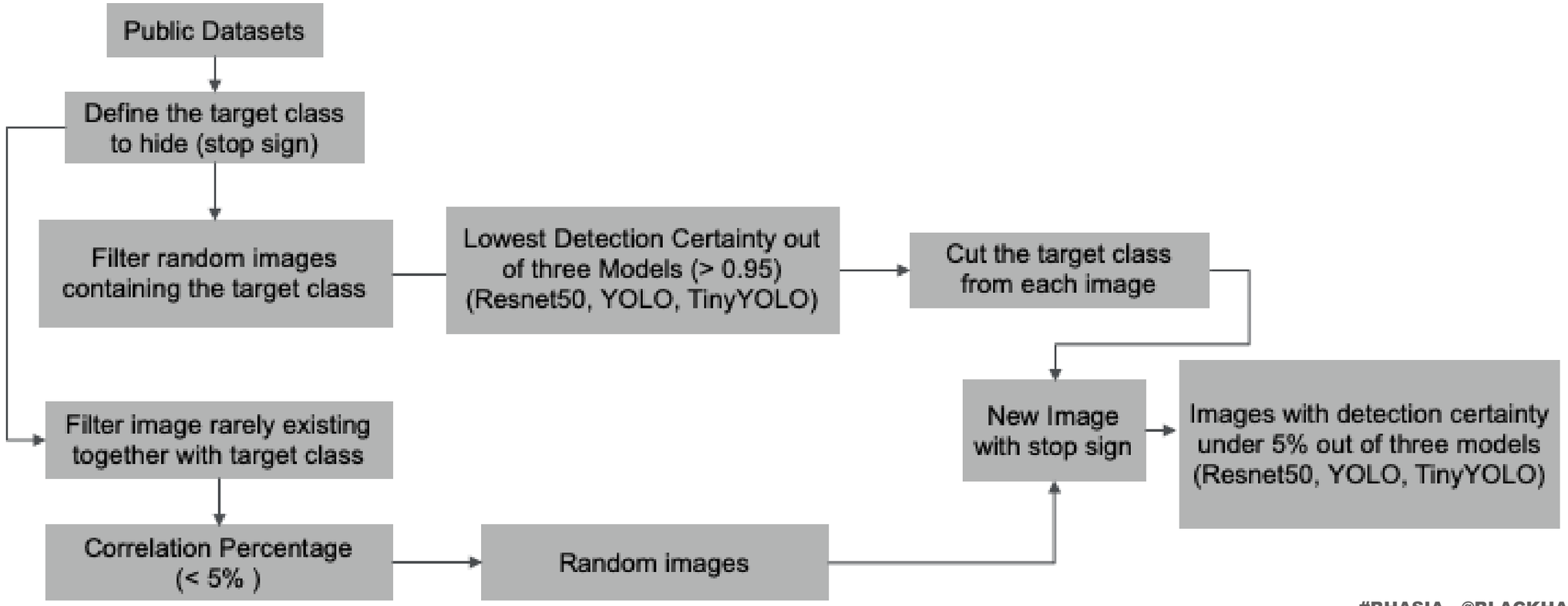
Automatic Generation Process

Step 2: Cut the target class from the image using its segmentation map and paste it onto a different image of a low-correlation class then re-detect to determine the certainty change.

COCO Dataset #570659

Automatic Generation Process

Step 3: Repeat this loop with pre-determined initial detection and adversarial detection percentages and save those where the target class is hidden.



Source Code Access:

https://github.com/DeloitteCyberSecurityLab/adversarial-image-generation

(MIT Licensed)

root@58b152158fa4: /data/exp

```
root@58b152158fa4:/data/experiments/adversarial# python run-loop.py 13
```

Examples of Images Hidden from YOLOv3 + COCO #1

COCO Dataset #508252

Examples of Images Hidden from YOLOv3 + COCO #2

COCO Dataset #501739

Examples of Images Hidden from YOLOv3 + COCO #3

COCO Dataset #260772

Adding More Computer Vision Networks

By adding more networks to the loop, the adversarial images become less extreme but more reliable.

COCO Dataset #113593

When testing images generated by running adversarial generation against YOLOv3 against a ResNet50 network, around half were hidden and the other half detected.

We added TinyYOLO and ResNet50 to the generation loop to create more universal images.

Phase 3: Testing Against Unrelated Commercial Networks

Since the correlation biases exploited in this approach exist in nature, they should be relatively consistent across all datasets and networks.

We ran 1,000 adversarial samples against major cloud vision AIs to test their detection rates.

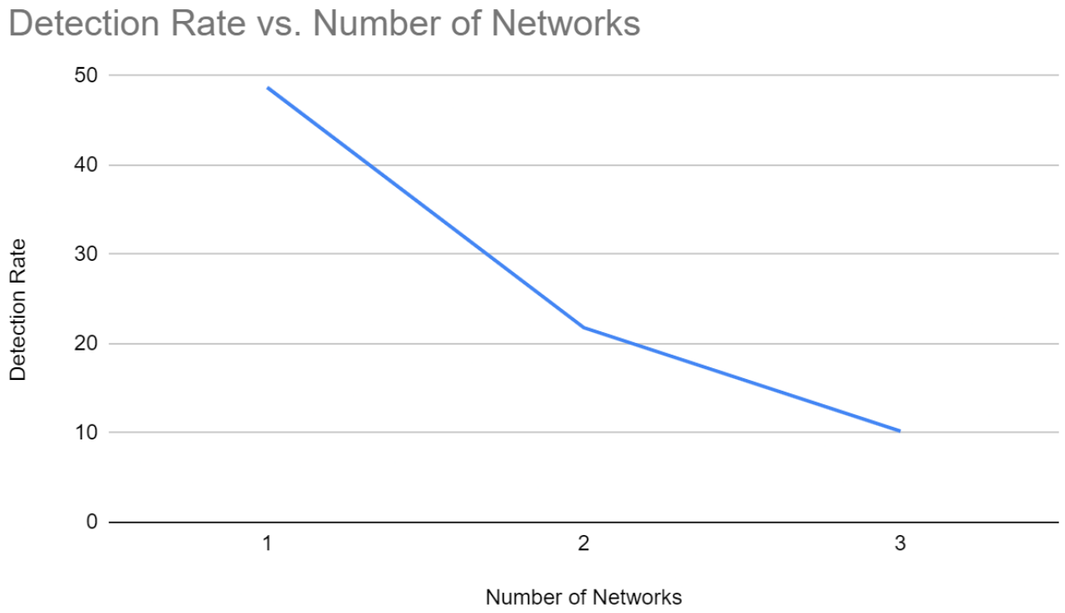
Detection Rates With Certainty Above 50% Across 1,000 Random Adversarial Images

Vendor	Detections	Average Certainty	Detection Percentage
A	102	74%	10.2%
B	68	57%	6.8%

While there were gaps in both the certainty and detection rate between the vendors, both systems were fooled by the majority of adversarial samples.

COCO Dataset #98029

Impact of Adding More Networks to the Generation Phase Measured Against Vendor A



Each network added in the generation step appears to roughly halve the detection rate of an unrelated network.

The lack of viable large-scale datasets prevented us from testing with more networks/datasets to see if this trend holds beyond 3.

Phase 4: Testing Against Physical Objects

So far, all tests were run against digital images.

However, the same approach should work for physical composite images.

Placing a physical stop sign onto a physical printout of a known adversarial background

Detection Rate >95% Across All Networks

Undetected Across 4/5 Networks

Since these attacks work in the physical domain, they can potentially be abused to attack computer vision in the wild.

E.g., self driving cars.

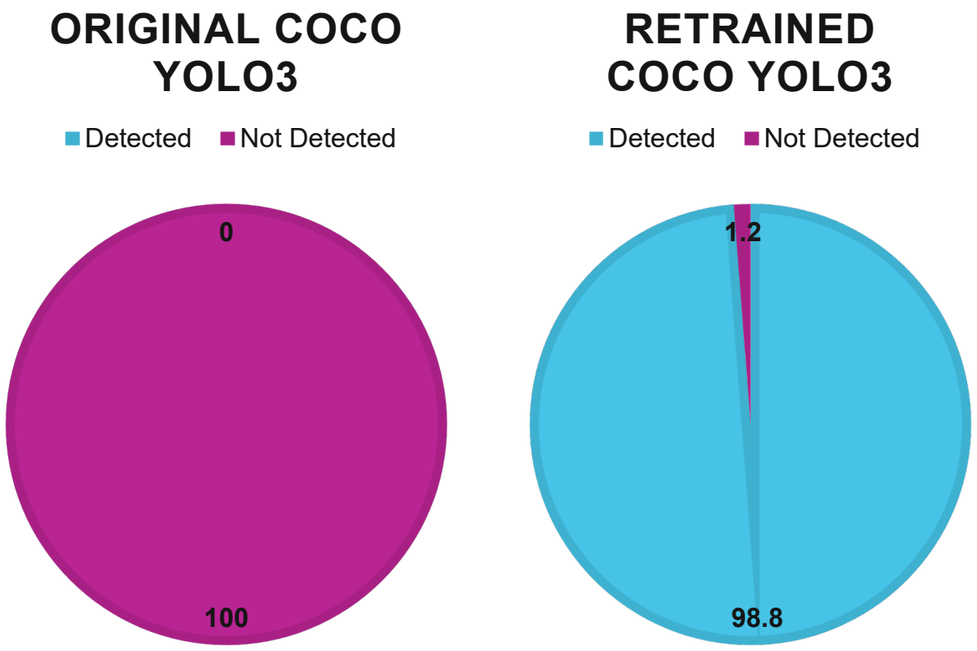
Mitigation

Since the vulnerability is caused by a bias in the dataset, it can be solved by amending the dataset.

Mitigation Approach

- Create a sample adversarial dataset with N = 1,492.
- Test the dataset against standard YOLOv3 COCO network (detection rate = 0.00%).
- Introduce 200 training and 70 validation samples from the adversarial dataset into the COCO dataset. The remaining 1,222 samples are used as the test group.
- Transfer train against the amended dataset for 2 hours.
- Run the new network against the 1,222 test images.

Mitigation Results



By adding only 200 samples, the attack was effectively mitigated.

Since COCO2017 has an original sample size of 118,288 samples, our required mix-in was only 0.16%.

Detection accuracy for the entire dataset was not impacted.

Original Mean Average Precision (MAP) : 57.8
Retrained MAP: 57.7
Standard MAP range: 55.0 – 58.0

Benefits of Correlation Based Attacks

Not Dataset-Specific or Network-Specific

The adversarial generation process does not have to be repeated for each new target network or dataset and should work well across new versions of datasets.

Works on Physical Objects

Deploying adversarial images as backgrounds pasted behind physical objects allows for attacks against physical infrastructure.

Can be Deployed Against Unknown Targets

Since the correlation biases exist in the real world, they are relatively universal across datasets.
This allows for attacks against unknown computer vision systems.

Target Object Does Not Have to be Obscured or Changed

Other approaches often require that at least a part of the target object is changed or covered by something else. In our approach, the target object remains unchanged.

Questions and Closing

Image Copyrights:
Dog: Helena Lopes – Pexels (NC)
Cat: Anel Rossouw – Pexels (NC)
Person: Luis Quintero – Pexels (NC)
Flower: Evie Schaffer – Pexels (NC)
Frisbee Dogs: COCO #358650, COCO #522818, COCO #352205