Egocentric Data as Natural Adversarial Examples

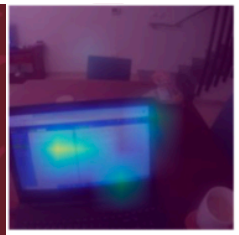
Mary Williamson

marywill@stanford.edu

CS231N (SCPD)

Introduction

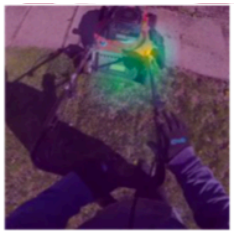
- Neural image classification models struggle to generalize to novel data
- Egocentric video or images from camera-wearers is becoming more common due to virtual/augmented reality applications
- I create "Ego4D Subset": a filtered dataset of ~3,300 static images from Ego4D egocentric video data
- I evaluate popular ImageNet-trained classifiers across 5 model families (convolutional and ViTs) both zero shot and with finetuning
- I find that zero shot performance is very low (~18% highest Top-1 Accuracy of any model)
- I finetune 5 models on this new data and find improved, but still low, performance (ViT had highest Top-1 Accuracy ~ 38%)



ResNet50
label: screen
conch (0.0049)
ladle (0.0047)
golf ball (0.0032)

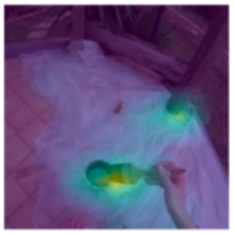
ResNet50
label: paintbrush
plunger (0.0048)
conch (0.0039)
syringe (0.0039)

ResNet50
label: mop
bucket (0.0053)
gong (0.0052)
spotlight (0.0043)



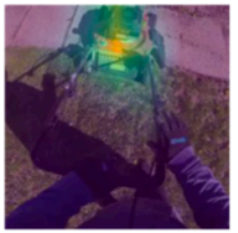
ResNet50
label: mower
lawn mower (0.0097)
stretcher (0.0084)
conch (0.0058)

ConvNext Base
label: screen
laptop (0.3289)
notebook (0.1379)
desk (0.0975)

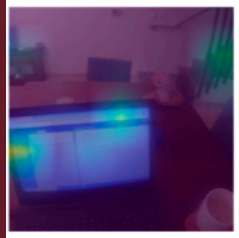


ConvNext Base
label: paintbrush
plastic bag (0.6608)
solar dish (0.0475)
rain barrel (0.0448)

ConvNext Base
label: mop
swab (0.8610)
broom (0.0211)
paintbrush (0.0055)



ConvNext Base
label: mower
lawn mower (0.8462)
harvester (0.0268)
swing (0.0040)



EfficientNetV2 b0
label: screen
plunger (0.0018)
paper towel (0.0017)
mailbag (0.0016)

EfficientNetV2 b0
label: paintbrush
binoculars (0.0019)
hammer (0.0018)
hook (0.0018)

EfficientNetV2 b0
label: mop
cellular telephone (0.0020)
hook (0.0017)
toilet seat (0.0016)

EfficientNetV2 b0
label: mower
sunglasses (0.0018)
hook (0.0018)
toy terrier (0.0017)

Related Work

- Neural Classifiers shown to struggle to generalize to novel data
- ImageNet-A dataset has "natural adversarial examples" that revealed common failure models in SOTA models without artificial images or noise
- Further work analyzing low performance on ImageNet-A: (1) multiple objects, (2) novel background, (3) small objects
- Other work shows models' over-reliance on texture and background and that they engage in "shortcut learning"

Reproduced from the ImageNet-A paper: Top: Categories *in* ImageNet that are classified wrong but with high confidence. Bottom: Classes not in ImageNet that are classified as ImageNet classes with high confidence.

Problem Statement

- Multiclass image classification: given an image, predict one of 1,000 classes (ImageNet-1K); 21,843 classes (ImageNet-21K)
- Input: Ego4D Subset of static images that I created from Ego4D Hands & Objects Video Dataset
- Model input images are 224x224x3 unless otherwise stated
- Metrics: Top-1 Accuracy or Top-5 Accuracy*
- Evaluations: Zero Shot and Finetuned on ~50% of my new Ego4D Subset (Validate on the other ~50% due to limited data)

*I implement "any of" multilabel classification accuracy for ImageNet-21K evaluations due to Ego4D annotation ambiguities

Dataset: Ego4D

- Summary
  - Egocentric video data (> 3,600 hours of video)
  - ~1,000 camera-wearers
  - 74 worldwide locations
- 5 benchmark tasks
  - Episodic Memory: visual/language queries (e.g. "where are my keys")
  - Hands & Objects: manipulation of objects in the hands
  - Forecasting: what will happen next?
  - A/V Diarization: localize the speaker
  - Social: who is talking to whom
- I use static images accompanying the Hands & Objects benchmark
  - Used for main dataset subtask to predict object states changes (e.g. an object burned, split, etc)

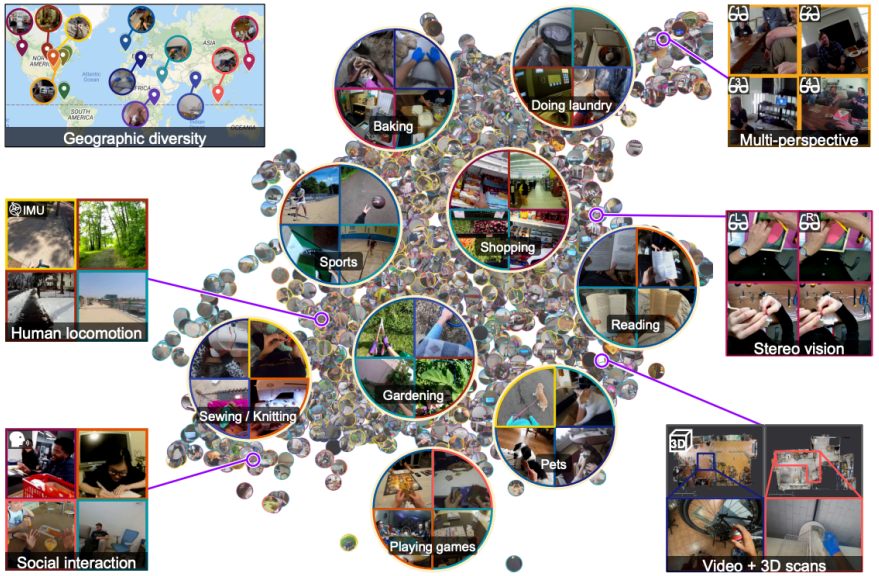


Figure 1. Ego4D is a massive-scale egocentric video dataset of daily life activity spanning 74 locations worldwide. Here we see a snapshot of the dataset (5% of the clips, randomly sampled) highlighting its diversity in geographic location, activities, and modalities. The data includes social videos where participants consented to remain unblurred. See https://ego4d-data.org/fig1.html for interactive figure.

Top Figure reproduced from Ego4D Paper

Ego4D Paper Statistics	
Hours of Data	196.2
Number of clips	88,585
Average length	8.0 sec
"Change Objects in Train Split"	19,347
Processed Dataset Statistics	
Total Clips Listed in Dataset Index	19,071
Duplicate Clip + Frame	107
Clip Had No Object Label	120
Total Parsed	17,754

Dataset

- Egocentric data is different!
  - Different perspective, unusual objects, objects are small, many objects in an image
- Aggressive filtering to create my Subset
  - Remove images with multiple objects
  - Remove non-ImageNet classes
- My new "Ego4D Subset"
  - 3,321 images for ImageNet-1K style pretrained-model evaluation
  - 7,398 images for ImageNet-21K style pretrained-model evaluation
- Imprecise Class Mapping
  - From Ego4D annotations to ImageNet-1K (e.g. "pot" vs "vase")
- ImageNet-21K Class Ambiguities
  - Had synonyms *in a single Ego4D annotation* that were distinct ImageNet-21K classes
  - Affected over half the examples (4,835 out of 8,488 images)
  - This is NOT: multiple annotations in the same image, which also was common
  - Example single annotation: "cloth(cloth,_fabric,_garment,_kanga,_rag)"

Figure 3. Examples from the Ego4D dataset that demonstrate its difficulty. Upper left: the mop is seen from an unusual perspective. Upper right: objects may be small relative to the size of the image. Bottom left: images may be cluttered with many objects. Bottom right: unusual tools and objects are common. Both the bottom conditions are filtered out of the new Ego4D Subset used in this work.

ImageNet 1K vs 21K	IN-1K	IN-21K
Total Images	17,754	17,754
Images with $\geq$1 IN class	3,474	8,488 [4,835]*
Images with $>$ 1 IN class	153	1,090
Invalid images	1	1
Images with 1 IN class	**3,321**	**7,398**
Total IN Classes	1000	21,483
Ego4D IN Classes	69	248 [419]*

Figure 5. Statistics of the processed Ego4D Subset when mapped to ImageNet-1K and ImageNet-21K classes [1]. *4,835 images have multiple distinct ImageNet-21K classes in a single annotation, which when included leads to 419 distinct classes in the Subset. If only the first ImageNet-21K class is included, there are 248 unique classes. Consequently, ImageNet-21K classification was phrased as an "any of" multilabel classification problem.

Methods

- Class Mapping
  - ImageNet-1K style: map Ego4D object labels to 1,000 ImageNet-1K classes
  - ImageNet-21K style: map Ego4D object labels to 21,843 ImageNet-21K classes
- Models evaluated
  - ResNet 50, 101, 152
  - RegNetY 8GF, 32GF
  - ViT B/16 and L/16
  - ConvNext Base, Large
  - EfficientNetV2 b0, b3, S, M, L
- Metrics
  - Top-1 & Top-5 Accuracy
  - For ImageNet 21-K due to class mapping ambiguities, I performed "any of" multilabel classification
- Finetuning Hyperparameters
  - Max 50 epochs (early stopping with 5 epochs patience)
  - SGD with Momentum (0.9)
  - Batch size: 128
  - Trained on 1 V100 or 1A100
  - Standard transformations
  - 3 LR warm-up epochs and decay 0.1/epoch
- Finetuning Data
  - Split Ego4D Subset in half by clip (not randomly by images) for training and validation
  - No 3rd test set due to limited images
  - If split randomly, there are too many similar shared images across train/val due to limited number of activities featured in the datasets
- Visualization & Analysis
  - GradCAM (Class Activation Mapping) of last layer
  - High confidence incorrect predictions above 0.90

Experiments [Zero Shot]

- Low performance
  - All models across the board
  - Highest EfficientNetV2 L 16.96 Top-1
- Larger model size helps somewhat
  - ViTs (Base 12% → Large 15% Top-1)
  - EfficientNetV2 (S 10% → L 16% Top-1)
- ResNet & RegNetY do the most poorly
- Models pretrained on ImageNet-21K, then finetuned on ImageNet-1K do slightly better than just pretraining on ImageNet-1K

ImageNet 1K-Trained Models on Ego4d Subset	Top-1	Top-5	Params (M)
ResNet50	5.42	12.02	25.6M
ResNet101	6.08	14.13	44.5M
ResNet152	6.17	13.77	60.2M
RegNetY 8GF	4.37	10.87	11.2M
RegNetY 32GF [288px]	7.98	17.59	19.4M
ConvNext Base	7.62	16.02	88.6M
ConvNext Large	7.89	17.56	197.8M
ViT B/16 [224px] (Alibaba)	9.28	17.44	86.5M
ViT B/16 [224px] (Google)*	12.41	24.4	86.5M
ViT L/16 [224px] (Google)*	15.87	**30.03**	304.3M
EfficientNetV2 b0 [224px]	3.95	10.75	7.1M
EfficientNetV2 b3 [300px]	10.31	20.87	14.4M
EfficientNetV2 S [384px]	10.3	20.72	21.5M
EfficientNetV2 M [480px]	11.75	20.81	54.1M
EfficientNetV2 L [480px]	**16.96**	27.98	118.5M

Figure 7. Zero shot Top-1 and Top-5 accuracy for models trained on ImageNet 1K. *Google ViTs were trained on ImageNet 21K and finetuned on ImageNet 1K. All models do poorly on this new dataset composed from Ego4D. However, larger models do have the best, albeit low, performance. The best Top-1 score is $\approx 17\%$.

Pre-training	ImageNet-1K vs ImageNet-21K Finetuned 1K	Top-1	Top-5
IN-1K	ViT B/16 [224px] (Google)*	9.28	17.44
IN-1K	ViT L/16 [224px] (Google)*	N/A	N/A
IN-1K	EfficientNetV2 L [480px]	**16.96**	**27.98**
IN-1K	ConvNext Large	7.89	17.56
21K-1K	ViT B/16 [224px] (Google)*	12.41	24.4
21K-1K	ViT L/16 [224px] (Google)*	15.87	30.03
21K-1K	EfficientNetV2 L [480px]	18.31	32.53
21K-1K	ConvNext Large	**18.73**	**32.68**

Figure 10. Top models are pretrained only on ImageNet-1K whereas bottom models are trained on ImageNet-21K and then finetuned on ImageNet 1K. You can see modest improvements across the board. ConvNext Large performance on ImageNet-1K versus the finetuned version has a larger jump than other models.

Experiments [Zero Shot]

- Same model architecture but larger input images helps slightly
  - Input resolution: 224px vs 384px
  - ViT Base/16: 12% → 15% Top-1
- Evaluated two of the large models pretrained only on ImageNet-21K
  - ImageNet-21K "any of" multilabel accuracy was very low

Vary Models Image Resolution on Ego4d Subset	Top-1	Top-5	Params (M)
ViT B/16 [224px] (Google)*	12.41	24.4	86.5M
ViT B/16 [384px] (Google)*	15.45	27.95	86.9M

Figure 8. Larger image input size (224px versus 384px) slightly improves accuracy for models of otherwise same architecture.

ImageNet-21K Trained	Top-1	Top-5	Params (M)
ConvNext Large [224px]	1.31	5.27	229.8M
EfficientNetV2 L [480px]	2.41	7	145.2M

Figure 9. These two models were trained only on ImageNet 21K, and the Ego4D Subset annotations were mapped to one of 21,483 classes. Classification was performed as an "any of" problem due to multiple distinct ImageNet 21K classes present in a single Ego4D annotation.

Experiments [Finetuning]

- Finetuned small model in each model family due to compute limitations
- Finetuning does improve model performance significantly
- Highest scores are RegNetY 8GF and ViT B/16 but still low < 40% Top-1
- ConvNext Base is the only model that does not improve significantly

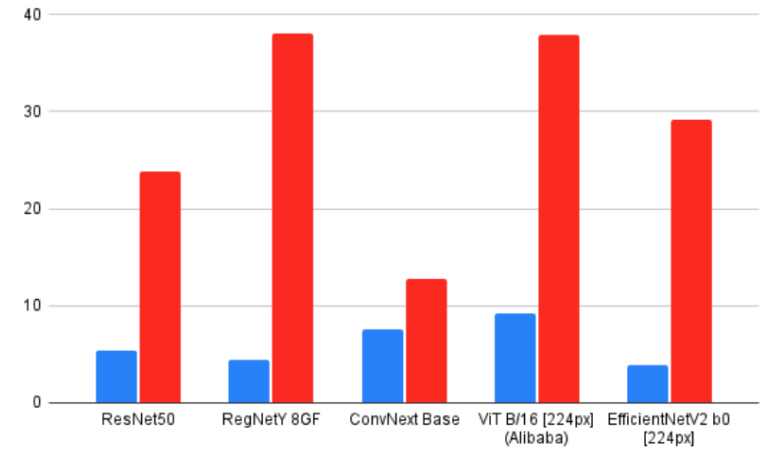


Figure 2. Original zero shot Top-1 accuracy on the Ego4D Subset compared with finetuned performance on 1,626 examples. Most models except ConvNext Base improved significantly, but even the best Top-1 accuracy is still relatively low at 37%.

Error Analysis

- Examined high-confidence incorrect predictions
- ResNet50 puts >0.90 probability where incorrect on 3.6% of all examples
- Error modes identified (See images at right)
  - My Ego4D Subset still has some images with multiple objects (top left: label: “vacuum”, prediction: seat belt”)
  - Egocentric perspective leads to unclear scenes overall (top right)
  - Imprecise class mapping (Bottom left: label: “vase”, prediction: “pot”)
  - Genuine errors (Bottom right: label: “wallet”, prediction: “cleaver”)
- Visualizing Errors
  - Using gradient-weighted class activation mapping (GradCAM) for last layer
  - Model focuses on the wrong areas (too much focus on background)
  - May focus on multiple areas of a single object instead of the whole object

Figure 6. High confidence ($p > 0.90$) but incorrect predictions by ResNet50 on zero shot evaluation. Top Left: label was “vacuum”, but model predicted “seat belt”, which is also in the image. Top Right: model predicted oxygen mask but label was “wool”. It is unclear what exactly that scene is. Bottom left: model predicted “pot” but label was “vase”. Bottom right: model predicted “cleaver” but label was “wallet”.

Conclusions

- I create the Ego4D Subset and evaluate various models both zero shot and after finetuning
  - "Ego4D Subset" consists of static images drawn from Ego4D egocentric video dataset
  - I evaluate common convolutional and ViT neural classifiers on this data
- Low performance on Ego4D Subset even after Finetuning
  - Existing models perform poorly zero shot on egocentric data (both convolutional and Vision Transformers) with top zero shot performance at ~18% top-1 accuracy
  - Larger size models in some families get some benefit
  - Small benefit to pretraining on ImageNet-21K first
  - Small benefit to input higher image resolution
  - Finetuning helps! But, top scores still low 38% highest Top-1 accuracy
- Error modes
  - Still having multiple objects in image (even after my annotation-based filtering)
  - Semantically similar class mapping differences (Ego4D to ImageNet)
  - Strange scenes due to egocentric perspectives (hard even for humans)
  - Genuine errors
  - Too much focus on background as per GradCAM visualizations
- Overall, this egocentric data is a promising for studying OOD!
  - Different distribution than traditional datasets is good for testing generalization capabilities of today's image neural classifiers
  - Today's models perform relatively poorly on this data

Future Work

- Improve mapping from one image class set (Ego4D) to another (ImageNet-1K)
- Improving multiple objects filtering (perhaps via cropping)
- Investigate other Ego4D annotations to see if more object data frames can be pulled
- Investigate object detection instead of classification
- Evaluate more models, bigger size; do more finetuning
- More error analysis on ImageNet-21K scores being low

References

- Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Q. Chavis, Antonino Furnari, Rohit Girdhar, Jack- son Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh K. Ramakrishnan, F. Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Z. Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Christian Fuegen, Abrham Gebre- selasie, Cristina Gonza lez, James M. Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Ja´chym Kola´r, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yang- hao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Mod- hugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz Puentes, Merey Ramazanova, Leda Sari, Kiran K. Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Yunyi Zhu, Pablo Arbela´ez, David J. Crandall, Dima Damen, Giovanni Maria Farinella, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard A. Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4d: Around the world in 3, 000 hours of egocentric video. ArXiv, abs/2110.07058, 2021.
- Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Xiaodong Song. Natural adversarial examples. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 15257–15266, 2021.
- Robert Geirhos, Jo¨rn-Henrik Jacobsen, Claudio Michaelis, Richard S. Zemel, Wieland Brendel, Matthias Bethge, and Felix Wichmann. Shortcut learning in deep neural networks. ArXiv, abs/2004.07780, 2020.
- Xiao Li, Jianmin Li, Ting Dai, Jie Shi, Jun Zhu, and Xiaolin Hu. Rethinking natural adversarial examples for classifica- tion models. ArXiv, abs/2102.11731, 2021.

*This is a non-comprehensive list!

Acknowledgements & References

- Entire project done by myself
- Relied on starter code and models from: pytorch-image-models, torchvision
- Important References
- Thank you to CS231N!