Exploring Robustness of Stanford's DeepSolar to Distribution Shifts

Kerrie Wu
Stanford University
kerriewu@stanford.edu

Julian Cooper
Stanford University
jelc@stanford.edu

Andrea van den Haak
Stanford University
vandenhaak@stanford.edu

Abstract

Robustness data distribution shifts is a topic within deep learning that has been well-studied from a theoretical standpoint, but is often not applied in applications that would benefit from it. In January 2023, Kasmi et al [6] published an aerial imagery dataset of residential areas in France, with classification labels and segmentation masks for solar panels in these images. In their Nature publication, Kasmi et al call attention to the inability of Stanford's Deepsolar model (which is trained on US aerial images) to generalize to this French dataset.

In this study, we explore two data-efficient techniques for improving the Deepsolar model's performance on the French dataset: regularized fine-tuning and interpolation-based data augmentation. We find that with just 500 target samples, our best model is able to achieve comparable performance on classification (F1 score of 0.94 vs 0.91) and segmentation (IOU of 0.48 vs 0.51) to the baseline Deepsolar model's performance on the original US dataset. However, additional error analysis suggests our model variants are unlikely to generalize to new domains without at least some minimal fine-tuning (e.g. 100 labeled target samples).

1. Introduction

We are interested in segmenting and classifying rooftop solar panels from aerial images, with a focus on residential areas. In many countries (including the United States), the rollout of rooftop solar has been ad hoc and largely untracked. Besides industrial sites and city buildings, electricity utilities are largely blind to where and how much solar neighborhoods have accumulated. Being able to estimate this (and its rate of increase) allows utilities to predict power output reasonably well (when combined with meteorology forecasts) and make the necessary adjustments to grid management and investment. [6]

In December 2018, Stanford's DeepSolar team produced a public aerial imagery dataset and model which together largely solved this challenge for the United States [16]. The original DeepSolar model has two branches, a classification branch that detects if there is a solar panel in an image, and a segmentation branch that produces a class activation map based on the image, which can be used to estimate the total area of solar panels in the image if the classification branch predicts positive [16]. However, even in its current form, the Deepsolar model is not robust to ”distribution shifts”. This means that the model performance drops when tested on datasets from regions with sufficiently different aerial imagery to its USA-based training data.

Earlier this year (January 2023), Kasmi et al published a new aerial imagery dataset focused on residential areas in France, with classification labels and segmentation masks for solar panels [6]. In their publication, Kasmi et al call attention to the inability of models trained on country-specific data to generalize to other regions [6]. We were able to confirm this hypothesis for Stanford's Deepsolar model, where the classification F1 score fell from 0.91 to 0.24 and segmentation IOU fell from 0.51 to 0.06 when testing on this new French dataset.

With the help of the DeepSolar team, we investigated different techniques for adapting the DeepSolar classification and segmentation model to a new domain, and potentially improving its overall robustness to distribution shifts. Our goal was to enhance the model's generalizability for use in countries other than the United States by requiring less labeled data from other countries for fine-tuning the model [16].

2. Related Work

In addition to studying the original publication from the Deepsolar team (Yu et al, 2018 [16]) for the US dataset, it was also instructive for us understand subsequent changes that were made in 2020 to adapt the model for a related application in Germany's most populous state, North-Rhine Westphalia [9]. In particular, this second implementation needed to handle input images of different size and resolution. We were able to use much of this logic when building our our adaptor to handle images from the French dataset for our finetuning task.

The most relevant publications for our project were different proposed techniques for handling distribution shifts for convolutional neural networks. First, Li et al [8] and Jiang et al [5] both explore how to fine-tune neural networks in a way that preserves much of the predictive power of the original model on the original (pretraining) dataset. Jiang introduces the SMART regularization technique which we use as inspiration for our fine-tuning routine. Second, Yao et al [15] introduced a data augmentation method called LISA which adds interpolations between original input-output training example pairs to the training data. The interpolation pairs can be selected in a targeted manner to improve robustness to domain shifts. Third, Csurka [2] provides a comprehensive survey of self-supervised and semi-supervised domain adaptation techniques up until 2017. Fourth, Hendrycks et al [4] describes a data processing technique which randomly generates augmentations and uses a Jensen-Shannon loss to enforce consistency. Fifth, Volpi et al [13] provides implementation details and analysis for how one might use Generative Adversarial Networks (GANs, originally introduced by Goodfellow et al [3]) to augment training data to improve robustness without introducing labeled samples from the target domain. At each training iteration, the training examples fed to the model are augmented with examples that are considered difficult for the current model. While we only had time to implement and properly test the first and second techniques from the above list, we hope to continue this work by exploring domain adaptation and GANs for this application (see Appendix 8.3).

We also reviewed several useful metrics for measuring "robustness to distribution shifts". Taori et al [12] provide the most recent overview of these metrics, including definitions for effective and relative robustness which we use to compare our model variants. The authors also prove that robustness to synthetic data distribution shifts does not imply robustness to natural distribution shifts (what we care about), and that a more diverse training set improves these measures of robustness as one would expect. Shankar et al [11] proposes another, more nuanced, metric of robustness: pm-k. This metric measures the top-k accuracy drop of a pretrained classifier when tested on a target domain dataset compared to the performance on the source domain dataset. We did not incorporate this into our analysis since there was not an intuitive k^{th} percentile for our particular application.

Finally, we briefly investigated augmenting the segmentation branch of the model to produce more accurate segmentation outputs by incorporating Facebook's Segment Anything Model [7]. The Segment Anything Model (SAM) is a foundational transformer vision model capable of zero-shot image segmentation, given an input image and point prompts of portions of the image to include, or exclude, in the mask. Our early experiments suggest that there might be potential to boost both segmentation IOU performance and resolution with intelligent prompting routines rather than expensive fine-tuning (further details in Appendix 8.2).

3. Methods

Before starting our analysis, we needed to invest significant work in being able to use the DeepSolar model with Pytorch [10], including (1) rewriting some components to port the model from TensorFlow into PyTorch, (2) fixing bugs associated with out-dated APIs, (3) ensuring code could natively make use of GPU resources, (4) rebuilding partially broken image processing pipelines, and (5) reproducing baseline numbers to convince ourselves that any fixes had been properly implemented.

To handle the distribution shift between our United States and France datasets, we experimented with regularized fine-tuning [8] and finetuning with interpolation-based data augmentation through LISA [15]. Based on [12]'s observation that a more varied training dataset results in better robustness, we were hopeful that the data augmentation with LISA would improve model robustness to distribution shifts. We also performed exploratory experimentation with techniques such as adversarial data augmentation [13] and Segment Anything Model [7] which is discussed in the appendices.

3.1. Regularized fine-tuning

Many fine-tuning routines suffer from overfitting, which leads to poor performance on test sets of downstream prediction tasks. Having carefully pre-trained our model on 400,000 aerials images of the rooftops across the United States, we do not want our finetuning to diverge too rapidly from the pre-trained weights. To control for this, we implemented (a) L2 regularization with tuned weight decay constant, and (b) adversarial regularization in our loss function.

To implement L2 regularization, we use the Adam optimizer and pass learning rate, beta and weight decay parameters. In particular, the weight decay constant guarantees we do not take too large steps away from our pretrained weights. (Note, this was effectively set to zero by the optimizer configured in the original Deepsolar model.) This tuning had a significant impact on our model's validation set performance and was replicated for our implementation of LISA (more detail in 3.2).

Jiang et al. proposed SMART regularization, which is an adversarial regularization technique built for finetuning Large Language Models. The desired property is that when the input x is perturbed by a small amount, the output should not change much. To achieve this, Jiang et al. [5] optimize loss $\mathcal{F}(\theta)$ using: $\min_\theta \mathcal{F}(\theta) = \mathcal{L}(\theta) + \lambda_s \mathcal{R}_s(\theta)$, where

$$\mathcal{L}(\theta) = \frac{1}{n}\sum_{i=1}^{n} l(f(x_i;\theta), y_i)$$

$$\mathcal{R}_s(\theta) = \frac{1}{n}\sum_{i=1}^{n} \max_{\|\tilde{x_i}-x_i\|_{p\leq\epsilon}} l_s\left(f(\tilde{x_i};\theta), f(x_i;\theta)\right)$$

As in [5], we use $l_s(P,Q) = \mathcal{D}_{KL}(P\|Q)+\mathcal{D}_{KL}(Q\|P)$ (symmetric KL-divergence). Here, the regularization term requires a maximization problem that can be solved efficiently using projected gradient ascent. Note that this regularization term is measuring the local Lipschitz continuity under the symmetrized KL-divergence metric. That is, the output of our model does not change much if we inject a small perturbation (constrained to be ϵ small in the p-euclidean metric) to the input. Thus, we can encourage our model f to be smooth within the neighborhoods of our inputs. While we had high hopes for this method to improve our model's robustness, we ultimately could not achieve meaningful predictions with this technique and so did not include in our reported results.

3.2. LISA data augmentation

LISA (Learning Invariant Predictors with Selective Augmentation) is a data augmentation method that uses linear interpolation between training examples to improve model robustness to distribution shifts [15]. We tried applying this method during finetuning. In the context of our specific experimental setup here, the algorithm for applying LISA during training is described below (closely adapted from the description in [15]. In this description, y is a vector label of size 2, with the first entry corresponding to the probability that the image is negative (contains no solar panel), and the second indicating the probability that the image is positive (contains a solar panel). For original, un-interpolated images, y will be a one-hot vector.

1. Select a training example and label X_1, y_1 from the French finetuning dataset.

2. Sample a λ value in the range (0, 1) from a $Beta(2,2)$ distribution.

3. Select a second training example and label X_2, y_2.

   - With probability p_{sel}, sample X_2 from the French finetuning dataset with label $y_2 \neq y_1$ (intra-domain interpolation).

   - With probability $1 - p_{sel}$, sample X_2 from the US finetuning dataset with label $y_2 \equiv y_1$ (intra-label interpolation).

4. Resize both X_1 and X_2 to the input size for the model.

5. Construct $X_{int} = \lambda X_1 + (1-\lambda)X_2$, and $y_{int} = \lambda y_1 + (1-\lambda)y_2$

6. Use X_{int}, y_{int} for training/finetuning.

We repeat these steps each time we sample from the finetuning dataset. Figure 1 shows a diagram with example images from one step of LISA data augmentation.

4. Dataset and Features

We make use of two satellite imagery datasets for this project: (1) the original Deepsolar United States dataset, published in 2017 [16], and (2) a recently published dataset for France (released January 2023) [6].

The United States dataset is large and highly skewed. It includes 46,090 images with solar panels (positive) and 366,467 images without solar panels (negative). For positive images, we also have access to masks with 1 and 0 pixels indicating exactly where in the image the solar panel lies. The dataset covers industrial and residential regions of the country. The resolution of each of the images is 320 x 320, as can be seen in Figure 2. [16].

The France dataset provides ground truth segmentation masks for 13,303 images from Google Earth25 and 7,686 images from the French national Institute of Geographical and Forestry Information (IGN). These two sources have substantial overlap and so we picked just one to focus on for this paper. We choose the Google-sourced images because they had a lower ground sampling distance (GSD) 0.1 meters per pixel (versus IGN of 0.2 meters per pixel), and hence lower resolution, as well as the fact that there were a larger quantity of them. The dimension of each of the Google and IGN images is 400 x 400 and they are both 72 pixels/inch.

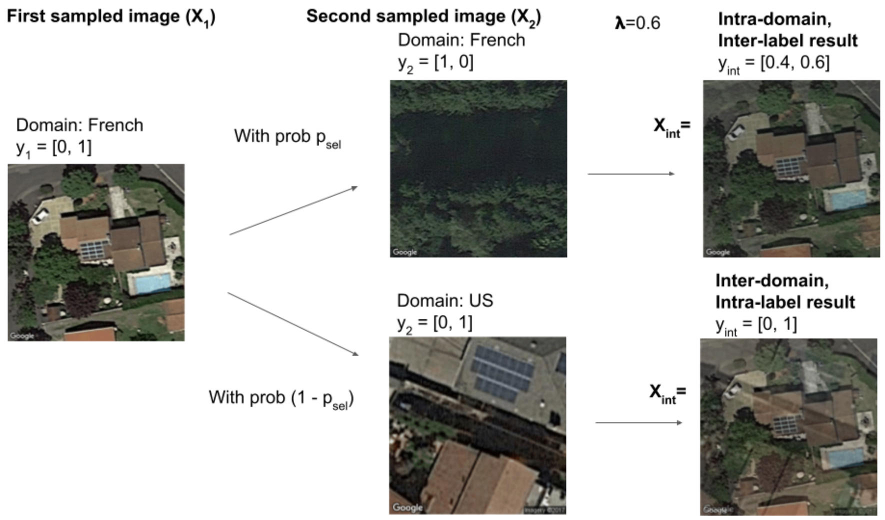
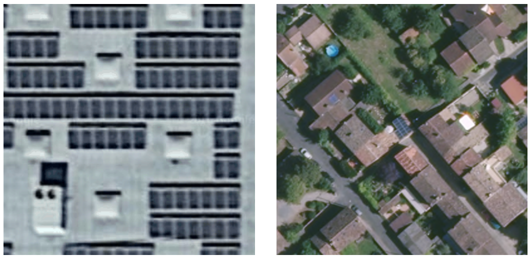


Figure 1. Diagram showing one step of LISA data augmentation.



(a) USA example (b) French example

Figure 2. Example images from our datasets

The data set focuses on more rural and residential regions versus the USA dataset. As can be seen in Figure 2. [6]

In addition to the LISA specific transforms covered in the methods section, we also applied random rotations in 90-degree increments, horizontal flips, and vertical flips. We resize the images to a standard input size of 299 × 299 and normalize the mean and standard deviation of each channel in the image for consistency as a preprocessing step. Apart from LISA, these data augmentation and preprocessing steps are the same as what was originally used for training the baseline DeepSolar model.

For the new approaches, we split the French data set into evaluation (5000 images per class), validation (1000 images per class), and training sets (the remainder of the images). We then sampled from the training set to create class-balanced finetuning (training) datasets with 100, 500, 1000, and 5000 images per class, respectively. For finetuning with LISA specifically, within each finetuning/training dataset, we also include an equivalent amount of data from each class from the US training dataset for interpolation/data augmentation. We used each of the small finetuning (training) datasets and the validation set for training and hyperparameter tuning the DeepSolar model, to try the two different finetuning approaches described in the Methods section. Finally, we evaluated all of our finetuned models and the baseline original DeepSolar model using both the original US DeepSolar evaluation set, and French evaluation set.

5. Experiments, Results, and Discussion

5.1. Model architecture and training pipeline

The DeepSolar model [16] consists of an Inception3 model for classification of images as containing a solar panel or not as the main backbone of the model. The auxiliary segmentation branch takes intermediate outputs of the Inception3 model as input and produces a 35 × 35 class activation map (CAM) as output. The CAM can be used to estimate the total area of solar panels in the image if the classification branch predicts positive. The segmentation branch is trained in a semi-supervised manner with a classification objective using the same input data and labels as the classification branch training data. A diagram of the architecture is located in the Appendix 8.1.

For each method and finetuning dataset, we performed finetuning in three steps, following the original training procedure used by the Deepsolar team [16]. This includes training the classification branch first, and then conducting greedy layer-wise training of the two-convolutional-layer segmentation branch.

1. Initialize the classification branch of the model with the pretrained, original DeepSolar model weights. Finetune the entire classification branch using the classification labels of the finetuning dataset.

2. Freeze the finetuned classification branch parameters, and initialize the first convolutional layer of the segmentation model from the original DeepSolar model parameters. Finetune the first convolutional layer of the segmentation branch using the classification labels of the dataset, attaching a global average pooling layer followed by a linear layer and softmax classification output to the convolutional layer.

3. Freeze the classification branch parameters and segmentation branch's first convolutional layer, and initialize the segmentation branch's second convolutional layer using the original DeepSolar model parameters. Finetune the second convolutional layer of the segmentation branch using the same classification objective and setup as in the previous step.

5.2. Evaluation metrics

The evaluation metrics we used for classification were precision, recall, and accuracy. Note that, although our training, validation, and evaluation sets from the French dataset are class-balanced, the US dataset is extremely class-imbalanced with the vast majority of samples being negative. Therefore, for metrics reported from the US dataset, accuracy is not a good measure of model performance–this is why we also utilize precision, recall, and F1 score, which are more indicative of true model performance on a skewed dataset.

For segmentation, we calculate an Intersection over Union (IoU) metric by comparing the predicted CAMs to the true segmentation masks, which we resize to the CAM size. This estimates how well positive areas of the CAM correspond to the true segmentation masks and allows us to interpret what the model activates on when classifying an image as containing a solar panel or not. We also report predicted area error, which is defined as the difference between predicted solar panel area and true solar panel area, divided by the true solar panel area. From a practical standpoint, this metric is useful when using model predictions to estimate the total solar panel area over a large quantity of satellite images. During finetuning, we use this metric to tune the segmentation threshold hyperparameter value for each model, described in the Hyperparameter tuning section below. Therefore, although we achieve low area errors via hyperparameter tuning, the IoU metric is more useful for understanding how well the model segments images and identifies areas corresponding to solar panels specifically.

To measure robustness, we use effective robustness metric$_{er}$ and relative robustness metrics metric$_{rr}$, adapted from the effective robustness and relative robustness metrics defined by [12]. The equations applied to our use case are defined below:

$$\text{metric}_{er} = \text{metric}(f, D_{fr}) - \text{metric}(f, D_{us}) \tag{1}$$

$$\text{metric}_{rr} = \text{metric}(f_{ft}, D_{fr}) - \text{metric}(f_{bs}, D_{fr}) \tag{2}$$

Within these equations, f represents a model, f_{ft} represents a finetuned model, f_{bs} represents the baseline original DeepSolar model, D_{fr} represents the French evaluation set, D_{us} represents the US evaluation set, and metric represents any metric that is scaled between 0 and 1, such as precision, recall, accuracy, or IoU.

Metric effective robustness measures how much better the model does on the French dataset compared to the US dataset. Metric relative robustness based on the French dataset measures how much better a finetuned model does on the French dataset than the baseline DeepSolar model. Together, they indicate how well a finetuned model adjusts to the distribution shift between US and French data. Ideally, a finetuned model will have close to 0 $metric_{er}$ indicating robustness to shifts between the French and US dataset domains (equal performance on each), and positive $metric_{rr}$, indicating improved robustness to the shift from US to French data compared to the original DeepSolar baseline.

5.3. Hyperparameter tuning

For regularized fine-tuning, we used grid search over learning rate, learning rate decay and L2 weight regularization (weight decay). Hyperparameter sweeps were done for each step of the fine-tuning pipeline (classification branch, segmentation branch convolution layer 1, and segmentation branch convolution layer 2) separately. For LISA data augmentation, we utilized Weights and Biases' [1] Bayesian hyperparameter tuning [14] feature to run sweeps. The tuned parameters included learning rate, learning rate decay, L2 weight decay, p_{sel}. Similarly to the regularized fine-tuning model, the hyperparameter sweeps were repeated for each step of the pipeline.

We selected hyperparameters based on the combinations that achieved the highest validation set classification accuracy. We chose to tune using the classification accuracy because our finetuning validation set is class-balanced, meaning that acccuracy provides a good overall, single-number representation of model performance. Tuning p_{sel} was only done for the classification branch finetuning. After classification branches were finetuned, the segmentation layers were finetuned using the same p_{sel} as the corresponding classification branch, based on the reasoning that changing p_{sel} could cause a change in the training data distribution and therefore negatively impact segmentation layer finetuning. However, we overall found that p_{sel} was not an important hyperparameter for tuning, and that learning rate, weight decay, and learning rate decay were much more important.

After finetuning the model with each method and finetuning dataset, we separately tuned the segmentation threshold value (between 0 and 1) for each finetuned model individually. The segmentation threshold value is used for identifying a positive pixel in the CAM produced by the segmentation branch. We select the value to the nearest hundredth's decimal place to minimize the percentage solar panel area error (described in the metrics section) predicted over the finetuning validation set.

5.4. Results

Table 1 provides a summary of our key experimental results, including classification and segmentation performance metrics for our finetuned model variants on the test French dataset. We train separate models for different amounts of labeled target data to understand the "efficiency" with which a particular finetuning method adapts to the new domain (i.e. does it need 100 or 5000 data points to achieve comparable results to USA baseline). Encouragingly, we are able to achieve comparable performance across classification and segmentation to the Deepsolar Baseline on USA dataset after finetuning on just 500 samples for regularized fine-tuning (e.g. IOU of 0.48 vs 0.51). While LISA performs similarly well for classification, it never quite achieves baseline performance for segmentation (IOU of 0.40 vs 0.51), but learns more quickly from the first 100 target samples and (as we'll see from the robustness analysis) is less prone to overfitting any given new domain.

Figure 3 visualizes how the performance of our two models changes as we increase the amount of available labeled target data across different metrics. As hoped, for classification accuracy we see a diminishing returns curve, with minimal uplift after 1000 data points. The story for segmentation is a bit messier. For relative area error, we quickly are able to achieve very good accuracy with minimal finetuning and so the differences after 100 data points are small and within margin of statistical noise. For IOU, both models again exhibit a diminshing retruns curve, notably with LISA outperforming Regluarized Finetuning for fewer labeled data points but converging asymptotically to a lower score. (For clarity we have removed the LISA IOU score for 1000 target data samples since we had signficiant trouble fitting this model as reflected by results in table 1.)

Table 2 results examine the ability of our models to generalize between the two datasets (domains). For classification, our finetuned models achieve good Relative Robustness (RR, want > 0) but poor Effective Robustness (ER, want ≈ 0). This suggests that while we have demonstrated our regularized fine-tuning and LISA data augmentation techniques can be finetune effectively (and efficiently) to new domains, we have not achieved a "robust" model that can handle multiple difference domains at once. One nuance is that our LISA model's ER (0.72 to 0.71) and RR (0.64 to 0.73) do not worsen as we increase the amount of label target samples we use for finetuning. This is an advantage of LISA over the regularized fine-tuning method for extrapolating to other domains without available labeled data. Note, we choose to use F1 score for this analysis since it combines precision and recall into one easy-to-compare metric and excludes accuracy which is misleading given the imbalance of the USA dataset (i.e. our French evaluation set is 50:50 while the USA dataset is 1:99 positive to negative samples).

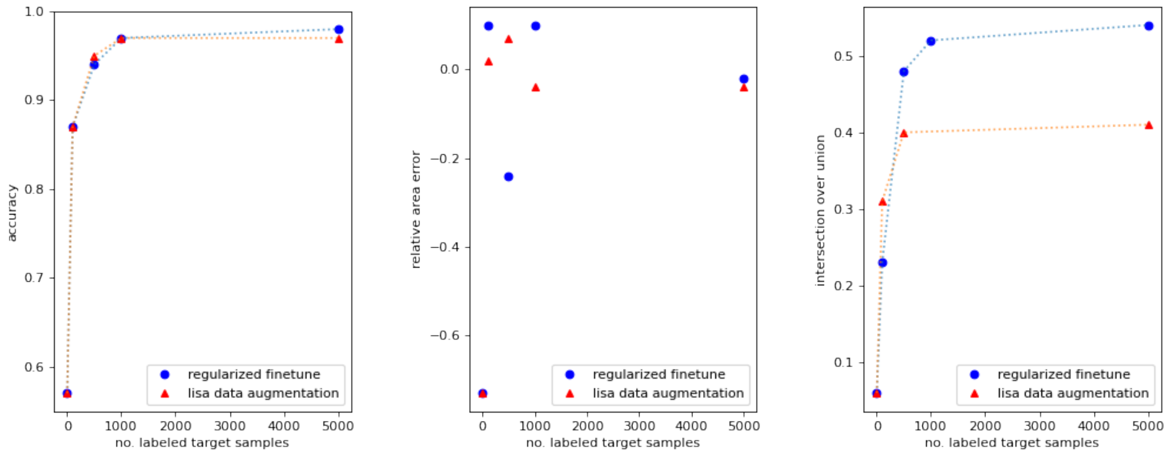
The story is similar for segmentation. Again we achieve good RR results across both models suggesting we have done a good job of adjusting to the French domain, but the ER results are still poor and confirm we probably cannot use these models without some minimal finetuning on new domains. We also observe that the LISA model's ER (0.28 to 0.32) and RR (0.25 to 0.32) hold up even as we increase the amount of target data used for finetuning, whereas regularized fine-tuning sees ER worsen significantly (0.20 to 0.52!). Note, we choose to use IOU for this analysis since it is a more precise (and less noisy) measure the quality of our segmentation task than relative area error.

5.5. Error Analysis and Examples

For cases where our classification accuracy was below 90% (i.e. when only 100 labeled target data samples used for finetuning) we wanted to understand the effect of our improvements made to the segmentation component independent of classification performance. To do this, we used an "oracle technique" whereby we fed the correct classification (labels) into the segmentation model and measured the difference in Area Error and IOU. We find that while the difference is negligible

			Classification				Segmentation	
Model	Eval. Dataset	Finetune Samples	Accuracy	F1 Score	Precision	Recall	Area Error	IOU
Deepsolar Baseline	USA	n/a	0.99	0.91	0.95	0.86	-0.08	0.51
Deepsolar Baseline	France	n/a	0.57	0.24	0.96	0.14	-0.73	0.06
Regularized Fine-tuning	France	100	0.87	0.87	0.86	0.89	0.10	0.23
Regularized Fine-tuning	France	500	0.94	0.94	0.97	0.91	-0.24	0.48
Regularized Fine-tuning	France	1000	0.97	0.96	0.96	0.97	0.10	0.52
Regularized Fine-tuning	France	5000	0.98	0.98	0.97	0.99	-0.02	0.54
LISA Data Augmentation	France	100	0.87	0.88	0.81	0.98	0.02	0.31
LISA Data Augmentation	France	500	0.95	0.95	0.95	0.96	0.07	0.40
LISA Data Augmentation	France	1000	0.97	0.97	0.96	0.98	-0.04	0.10
LISA Data Augmentation	France	5000	0.97	0.97	0.96	0.98	-0.04	0.38

Table 1. Summary of experimental results showing performance of (1) the Deepsolar Baseline model (pretrained) on the original US dataset compared to results with the same model on the French dataset; and (2) the Regularized Fine-tuning and LISA Data Augmentation model variants trained on varying quantities of labeled data from the target (French) datatset. Accuracy, Precision and Recall are all based on the image classification performance of the model. Area error measures the relative difference between the predicted solar PV area and the true solar PV area as measured by the true segmentation masks. IOU stands for intersection over union which is our primary measure of correctness for the segmentation task.



(a) Accuracy

(b) Relative Area Error

(c) Intersection over Union

Figure 3. Compares convergence of both models across several key metrics as we increase available labeled target data.

for finetuning with 500 or more labeled target samples, the difference is meaningful for IOU when only exposed to 100 samples from target distribution (0.23 vs 0.28 and 0.31 vs 0.38). This matches our intuition that the classifier performance has a significant flow through impact on segmentation. This is particularly true for recall (otherwise segmentation model never gets a chance to predict pixels!) and helps to explain why LISA outperformed regularized fine-tuning in the 100 data samples scenario. Interestingly, we did not observe this relationship for Area Error, however, this was overall a much noisier (and low overall error) metric.

Figure 4 shows five illustrative examples of where our models succeed and fail for segmentation. The first two rows capture the main difference between the original Deepsolar Baseline and our fintuned model variants: misclassification of images flowing through to poor segmentation. Our baseline model, for example, achieved precision of 0.96 but recall of only 0.14. This means it was confident in the images it did select but misclassified a bunch of images that did in fact have solar panels. For those images, it predicted an empty

		Classification: F1 Score				Segmentation: IOU			
Model	Finetune Samples	F1(US)	F1(Fr)	ER	RR	IOU(US)	IOU(Fr)	ER	RR
Deepsolar Baseline	n/a	0.91	0.24	-0.67	n/a	0.51	0.06	-0.45	n/a
Regularized Fine-tuning	100	0.22	0.87	0.65	0.63	0.03	0.23	0.20	0.17
Regularized Fine-tuning	5000	0.06	0.98	0.92	0.74	0.02	0.54	0.52	0.48
LISA Data Augmentation	100	0.16	0.88	0.72	0.64	0.03	0.31	0.28	0.25
LISA Data Augmentation	5000	0.26	0.97	0.71	0.73	0.06	0.38	0.32	0.32

Table 2. Calculates "robustness distribution shift" metrics suggested by [12], including Effective Robustness (ER) and Relative Robustness (RR). We show these results for classification F1 score and segmentation IOU for Deepsolar Baseline, Regularized Fine-tuning and LISA Data Augmentation models at 100 and 5000 labeled target data samples.

	Area Error		IOU	
Model	Classifier	Oracle	Classifier	Oracle
RF	0.10	0.12	0.23	0.28
LISA	0.02	-0.19	0.31	0.38

Table 3. Comparing segmentation metrics (Area Error, IOU) for Regularized Fine-tuning and LISA Data Augmentation models when we use an oracle vs trained classifier as input. All experiments in this table as for finetuning with 100 labeled target samples.

segmentation! This clearly flows through to our baseline segmentation results where we significantly underpredict total area (-73%) and achieve only 0.06 IOU.

The subsequent three rows in Figure 4 show more nuanced cases where we succeed and fail. Our largest sources of error remaining in the regularized fine-tuning and LISA data augmentation models were due to diffuse prediction (row 3) and object confusion (row 4). As we add more data from the French dataset which is more heavily weighted towards residential settings, our models improved at correctly identifying objects (e.g. row 5 where we no longer confuse a swimming pool for a solar panel) but do not improve much in terms of diffuse prediction error (coloring in panel plus remaining roof space).

To address diffuse prediction error, we also tried tuning our CAM segmentation threshold (how confident do we need to be for a pixel to be considered 1 vs 0) as an additional hyperparameter. This made a significant difference for relative total area error but we found negligible uplift for IOU since our model will often put comparable weight on panel and roof so we could easily be eliminating some of the true pixels as we remove false pixels.

6. Conclusion

In conclusion, this paper presented an analysis of the DeepSolar model's performance on aerial imagery datasets from the United States and France. Deepsolar suffers significant performance loss when evaluated on the French dataset, with classification F1 score falling from 0.91 to 0.24 and segmentation IOU falling from 0.51 to 0.06. To overcome the distribution shift between the United States and France datasets, two approaches were explored: regularized fine-tuning and LISA data augmentation.

We find both approaches achieved comparable classification performance to Deepsolar on the US dataset after training on just 500 samples (F1 scores of 0.94 and 0.95 vs Deepsolar's 0.91). Similarly for segmentation, both models demonstrated significant uplift after 500 samples (IOU of 0.48 and 0.40 vs Deepsolar's 0.51). While these results confirm we are able to effectively (and efficiently) finetune our original model weights for the new French dataset, an analysis of robustness metrics suggests that our model variants are unlikely to generalize to new domains without at least some minimal fine-tuning.

7. Future Work

Future work should include expand this investigation to include additional known distribution shift techniques such as Domain Adaptation and General Adversarial Networks (GANs). It would also be interesting to investigate if choosing a lower p_{sel} value when finetuning with LISA could improve model robustness to distribution shifts, because a lower p_{sel} value encourages using more diverse domain samples during finetuning–although we tuned p_{sel} within this project, we tuned to maximize performance on the French dataset rather than for model robustness metrics.

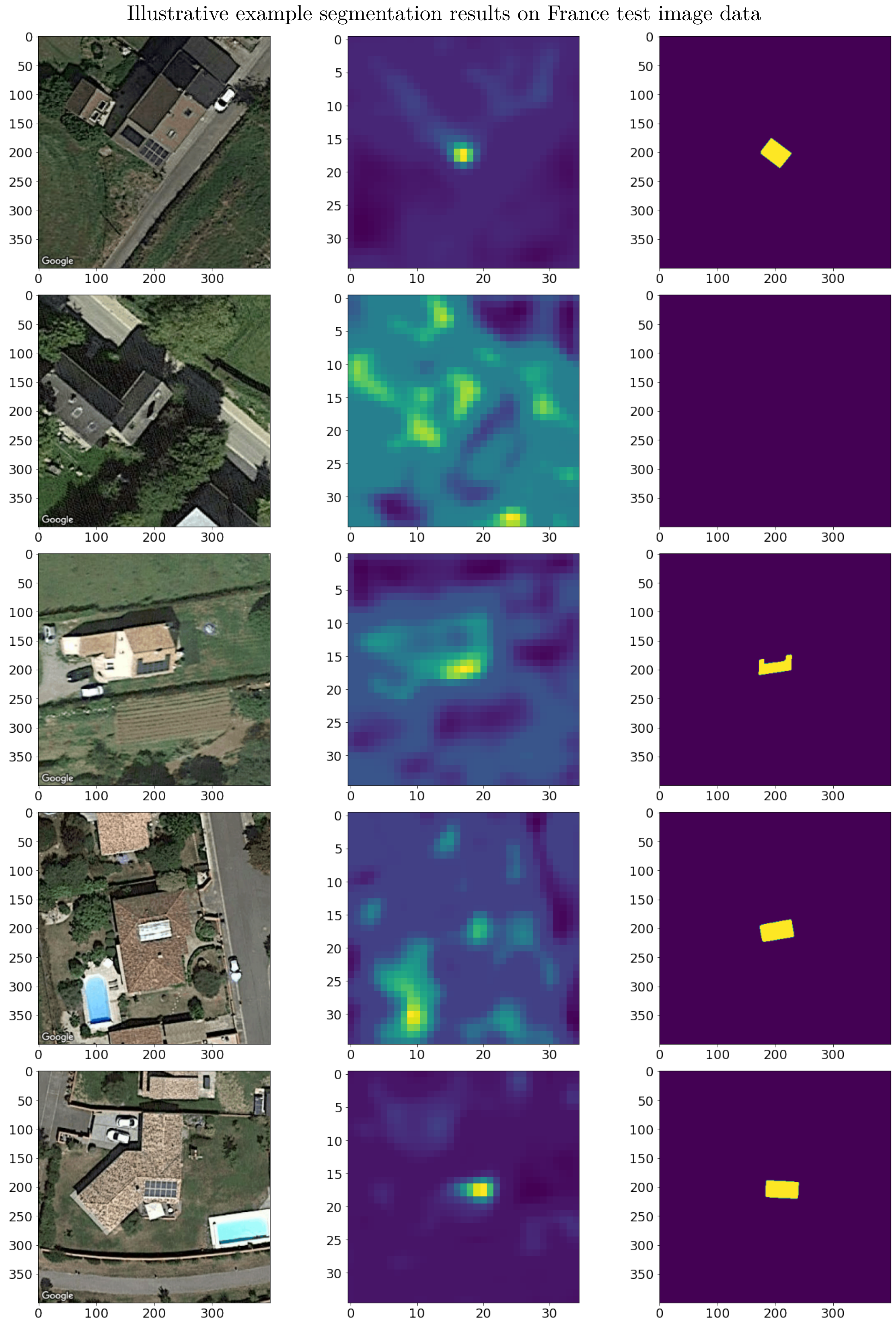

Illustrative example segmentation results on France test image data
0
50
100
150
200
250
300
350
0
100
200
300
0
5
10
15
20
25
30
0
10
20
30
Google

Figure 4. Columns: original input image (left), CAM activations from Regularized Fine-tuning model (middle), and true segmentation masks (right). Rows: (1) good example of a simple one panel segmentation; (2) bad example of model misclassification; (3) bad example of overly diffuse prediction; (4) bad example of confusing panel for another object in the scene; (5) good example of correctly ignoring that same object (backyard swimming pool).

8. Appendices

8.1. Deepsolar Model Architecture

See Figure 5 for Deepsolar model architecture, including Inception3 classification and two levels of segmentation.

8.2. Segment Anything Model

During the first month of the project we experimented with Facebook's newly released Segment Anything Model (SAM) [7]. This was a suggestion by the Deepsolar team to see if we could boost segmentation performance on the original USA dataset. Our initial experiments did show some promise! By randomly sampling three positive and three negative points from our predicted segmentation CAM and passing these as prompts to the SAM model, we were able to achieve 0.47 relative area error and 0.24 IOU. We ultimately choose to focus our attention on the distribution shift problem since we were more excited about it's potential use cases beyond this course, but would suggest there is certainly further potential in designing more sophisticated prompts to improve the Deepsolar + SAM segmentation results.

8.3. Generative Adversarial Network

We also experimented with developing a Generative Adversarial Network (GAN) to enhance the training data for the satellite imagery models based on some of the recommendations in [13]. The objective was to create a more robust and accurate model by augmenting the training examples with challenging samples.

At each iteration of the training process, we tried to incorporate difficult examples that were specifically chosen to challenge the current model. This approach aimed to improve the model's ability to handle complex and diverse scenarios, ultimately leading to better performance on the distribution shift issues experienced by the Deepsolar model between the French and the USA datasets.

While experimenting on AWS, after the first iteration, the Discriminator (D) and Generator (G) losses were measured at 1.385 and 0.5776, respectively. This indicated that the model was learning and making progress towards achieving the desired outcome.

Unfortunately, the process was interrupted or terminated after this initial iteration, resulting in incomplete results. Further analysis and experimentation are needed to draw definitive conclusions about the effectiveness of this approach for the Deepsolar model, and so we recommend this larger body of research for future work.

9. Contributions and Acknowledgements

Github code repository for this project can be accessed at https://github.com/jelc53/deepsolar.git

Contributions: Kerrie owned development of the LISA data augmentation model implementation and experimented with SAM applications. Julian owned development of the regularized fine-tuning model implementation and experimented with GAN applications. Andrea led the literature review, owned data processing pipeling for the French dataset, and experimented with GAN applications. Everyone contributed to the model setup, running experiments on AWS server, and brainstorming design choices.

Acknowledgements: We would like to thank Zhecheng Wang for guidance on initial directions for the project, introducing us to DeepSolar Github repositories that we built upon, and providing us the original DeepSolar US dataset.

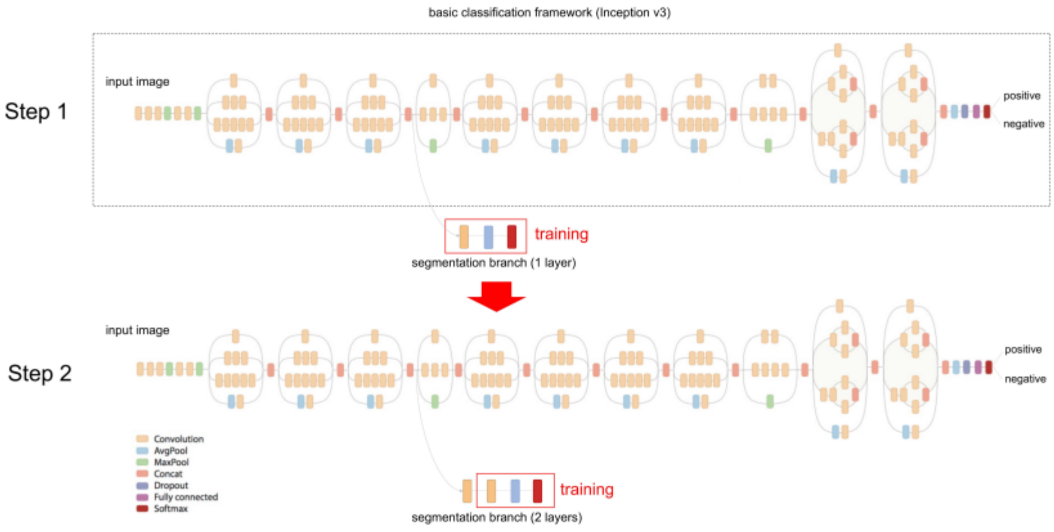


Figure 5. The layers in the dashed box form the basic classification framework (Inception-v3 model). Greedy layer-wise training is performed in two steps (labelled in the left in the figure). Step 1: Keep all layers fixed but train a single "convolutional layer-GAP-linear classifier" structure in the segmentation branch. Step 2: Add another "convolutional layer-GAP-linear classifier" structure at the end of the segmentation branch and train it with all other layers fixed. [16]

References

[1] Lukas Biewald. Experiment tracking with weights and biases, 2020. Software available from wandb.com. 5

[2] Gabriela Csurka. Domain adaptation for visual applications: A comprehensive survey. CoRR, abs/1702.05374, 2017. 2

[3] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014. 2

[4] Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty, 2020. 2

[5] Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. SMART: robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization. CoRR, abs/1911.03437, 2019. 2, 3

[6] Gabriel Kasmi, Yves-Marie Saint-Drenan, David Trebosc, Raphaël Jolivet, Jonathan Leloux, Babacar Sarr, and Laurent Dubus. A crowdsourced dataset of aerial images with annotated solar photovoltaic arrays and installation metadata. Scientific Data, 10(1):59, Jan 2023. 1, 3, 4

[7] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023. 2, 10

[8] Dongyue Li and Hongyang R. Zhang. Improved regularization and robustness for fine-tuning in neural networks, 2021. 2

[9] Kevin Mayer, Zhecheng Wang, Marie-Louise Arlt, Dirk Neumann, and Ram Rajagopal. Deepsolar for germany: A deep learning framework for pv system mapping from aerial imagery. pages 1–6, 2020. 2

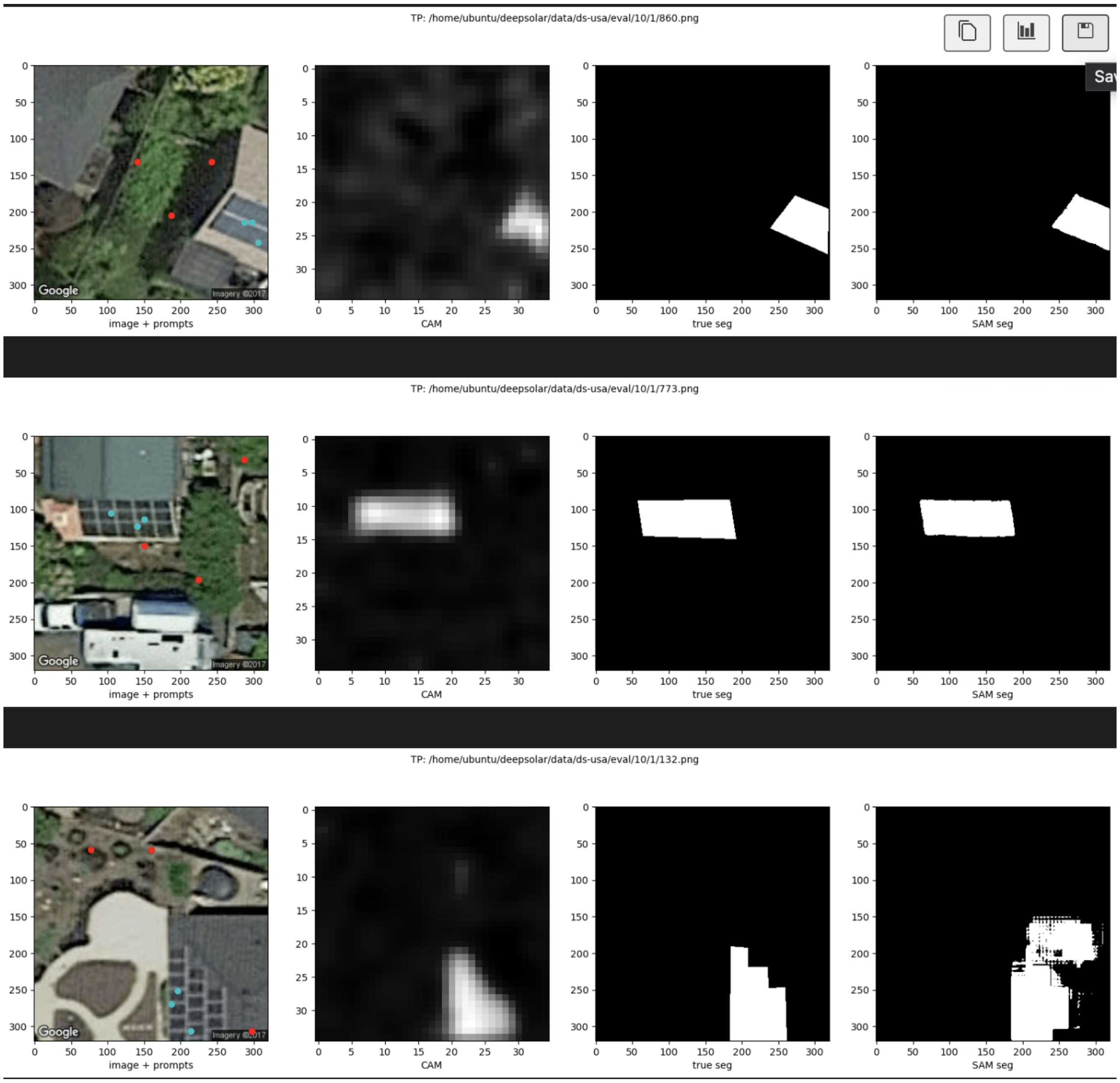
[10] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Advances in Neural Information Processing Systems 32, pages 8024–8035. Curran Associates, Inc., 2019. 2

[11] Vaishaal Shankar, Achal Dave, Rebecca Roelofs, Deva Ramanan, Benjamin Recht, and Ludwig Schmidt. Do image classifiers generalize across time?, 2019. 2

[12] Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. Measuring robustness to natural distribution shifts in image classification. CoRR, abs/2007.00644, 2020. 2, 5, 8

[13] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation, 2018. 2, 10

[14] Jia Wu, Xiu-Yun Chen, Hao Zhang, Li-Dong Xiong, Hang Lei, and Si-Hao Deng. Hyperparameter optimization for machine learning models based on



TP: /home/ubuntu/deepsolar/data/ds-usa/eval/10/1/860.png

TP: /home/ubuntu/deepsolar/data/ds-usa/eval/10/1/773.png

TP: /home/ubuntu/deepsolar/data/ds-usa/eval/10/1/132.png

Figure 6. Preliminary segmentation of solar PV cells with SAM (rightmost column), compared to true segmentation masks (second from right), CAM activations from the original DeepSolar model (third from right), and the original input image (leftmost) with positive point prompts in cyan, and negative point prompts in red.

bayesian optimizationb. Journal of Electronic Science and Technology, 17(1):26–40, 2019. 5

[15] Huaxiu Yao, Yu Wang, Sai Li, Linjun Zhang, Weixin Liang, James Zou, and Chelsea Finn. Improving out-of-distribution robustness via selective augmentation, 2022. 2, 3

[16] Jiafan Yu, Zhecheng Wang, Arun Majumdar, and Ram Rajagopal. Deepsolar: A machine learning framework to efficiently construct a solar deployment database in the united states. Joule, 2(12):2605–2617, 2018. 1, 3, 4, 11