Abstract

Many vision tasks use secondary information at inference time—a seed—to assist a computer vision model in solving a problem. For example, an initial bounding box is needed to initialize visual object tracking. To date, all such work makes the assumption that the seed is a good one. However, in practice, from crowdsourcing to noisy automated seeds, this is often not the case. We hence propose the problem of seed rejection—determining whether to reject a seed based on the expected performance degradation when it is provided in place of a gold-standard seed. We provide a formal definition to this problem, and focus on two meaningful subgoals: understanding causes of error and understanding the model’s response to noisy seeds conditioned on the primary input. With these goals in mind, we propose a novel training method and evaluation metrics for the seed rejection problem. We then use seeded versions of the viewpoint estimation and fine-grained classification tasks to evaluate these contributions. In these experiments, we show our method can reduce the number of seeds that need to be reviewed for a target performance by over 23% compared to strong baselines.

1. Introduction

Many tasks in computer vision require not only a primary input, such as an image or a video, but also additional information based on the primary input—a seed—to be provided to the task model. This seed may be used to define the problem, such as in visual object tracking [28], video object segmentation [36], and visual question answering [1], or to provide additional information for common tasks such as fine-grained scene classification [27], visual concept prediction [57], or viewpoint estimation [52]. Critically, these tasks are evaluated using verified gold-standard seeds, ignoring the noisy processes by which seeds are generated.

The performance of computer vision models with poor primary inputs has been explored in the context of naturally difficult [55, 69, 9, 14] and intentionally adversarial [58, 61, 10, 51] primary inputs, leading to a variety of methods designed to make models more robust [55, 69] or detect and reject difficult inputs [14]. However, no work to our knowledge has been performed on the identification and rejection of bad seeds: seeds that cause a significant increase in error on the task when used in place of the gold-standard seed. As reliability issues in crowdsourcing are well studied [24, 39, 48, 44] and automated systems that could be used to create seeds are subject to unpredictable failure modes [42, 61], not having any mechanism for detecting bad seeds is a critical oversight.

To emphasize the need for such a mechanism, we examine Figure 1, where a human annotator is asked to click a semantically meaningful location on the image (e.g. rear seat) to resolve the viewpoint estimation model’s perceptual ambiguities. This example illustrates the complex, and sometimes counterintuitive, interaction between the primary input, seed, and task model: while many seeds that are incorrect in the input space (e.g. the yellow seed) don’t de-
Figure 2: On both the KCVE (top row) and HSC (bottom row) tasks, the task model may or may not condition its answer solely on the primary input. For KCVE, the gold-standard seed is shown as a green circle, while the overlaid heatmap shows error from low (green) to high (red). For HSC, the gold-standard seed is in bold, correct answers are shown in green, and incorrect answers are shown in red.

gain an understanding of the task model’s response, and how a human’s intuition of a seed’s quality differs from its effect on the accuracy of the task model’s output. We again highlight the example shown in Figure 1, where a small Euclidean error in the input space (red keypoint) can cause a large increase in output error, while a much larger Euclidean error (yellow keypoint) may have little effect.

To address these challenges, we propose Dual-loss Additional Error Regression (DAER), a novel training method developed for the seed rejection problem. DAER considers the two challenges discussed above separately during training, and combines them during inference to predict the effect of a candidate seed on the downstream task. We evaluate the performance of DAER on two tasks: keypoint-conditioned viewpoint estimation [52]—a human-in-the-loop extension of the canonical viewpoint estimation task [54, 50, 68, 35, 32]—and hierarchical scene classification [27]—a method that improves performance on fine-grained classification [56, 67, 31, 63] by integrating a coarse scene classification.

To evaluate DAER, we introduce a task-agnostic benchmark evaluation method for seed rejection, centered around new metrics designed specifically to assess the performance of a seed rejection method: Additional Error (AE), Mean Additional Error (MAE), and Area under the Mean Additional Error curve (AMAE). Unlike existing metrics, such as selective risk [15], these metrics focus on the potential benefit of a new seed, instead of an oracle label of the target value that may be prohibitively difficult to obtain at scale.

The contributions of this paper are as follows:

1. A formalization and benchmark metrics for the seed rejection problem, in which a model is tasked with determining if a candidate seed will produce significantly higher error than the corresponding (unknown at inference time) gold-standard seed.
2. Dual-loss Additional Error Regression (DAER), a broadly applicable training and inference method for the task of seed rejection.
3. An evaluation of DAER on the tasks of keypoint-conditioned viewpoint estimation [52] (KCVE) and hierarchical scene classification [27] (HSC), which shows that DAER can reduce the the number of seeds that need to be reviewed for a given target performance by over 23% compared to the best-performing baseline.

2. Related Work

2.1. Seeded Inference

Seeded inference describes a number of problems in which a task model accepts a primary input and additional information based on that primary input—a seed—and estimates a target value. Though the list of problems that can be classified as seeded inference is long [47, 4, 49, 16, 41, 57, 56],
While some work acknowledges that performance can be improved by choosing which seed to request [2, 17], current work generally does not consider the seed itself to be subject to error. In cases where the seed is categorical, such as hierarchical scene classification [27, 57], seeds other than the gold-standard are not considered. In contrast, many works in which the input space is effectively continuous, such as keypoint clicks in keypoint-conditioned viewpoint estimation [52] and bounding boxes in visual object tracking [28, 60], acknowledge that seeds can be noisy and either seek to improve robustness [45], or simply evaluate the robustness of existing models to a range of expected noise defined a-priori [60, 52]. Critically, in addition to ignoring the effect of seeds that are not within this predefined range, these methods do not consider which specific seeds result in an increase in error.

2.2. Selective Prediction

A problem closely related to seed rejection is the problem of selective prediction [6, 14]. In selective prediction, the goal is to split primary inputs into a set that is classified by a task model and a set that is classified by expert human annotators such that annotation cost is minimized subject to an error constraint, or error is minimized subject to a cost constraint. Selective prediction has been applied to many regression and classification strategies over time, from nearest neighbors in the 1970’s [20], to support vector machines in the early 2000s [12], to deep artificial neural networks today [64, 14]. Gurari et al. [18] extend the problem of selective prediction by considering the case where multiple models (including human annotators) are available, and predicting a best performer based on a regressed intersection-over-union.

While both selective prediction and seed rejection predict performance of a task model on a given input, selective prediction only considers rejection of a single input, which would be seen in tasks such as as image classification [13, 38, 15] or tabular regression [53, 14]. In these tasks, the only option if the primary input is rejected is to receive a target label from a human expert. This results in an unnecessary increase in annotation cost due to the target label being inherently more difficult to obtain than a seed. For example, it is substantially easier to perform a keypoint click than a full viewpoint annotation [52], or to initialize an object tracker with a first-frame bounding box than draw a bounding box on every video frame [28].

3. Seed Rejection

Here, we first define seed rejection and its associated metrics in a problem-independent manner, where task and rejection models may be parameterized by learned or hard-coded methods. Next, we present a generic formulation of our proposed solution, which we call Dual-loss Additional Error Regression (DAER). In Section 4, we instantiate this methodology in two concrete problems.
3.2. Aggregate Metrics

With the goal of a rejection model defined, we note that aggregate metrics are required for parameter tuning and comparing the performance of rejection models on a test set, $\mathcal{D}$. We hence propose the Mean Additional Error (MAE), which corresponds to the mean of all additional errors across an accepted set of samples:

$$
\text{MAE}(f, g|\mathcal{D}, \ell) = \frac{1}{|\mathcal{D}|} \sum_{(x, s_c, s_{gs}, y) \in \mathcal{D}} g(x, s_c) \Delta E(x, s_c, s_{gs}, y|f, \ell).
$$

Since a target coverage or MAE is chosen based on an application constraint (e.g., budget) we further seek a metric that can compare rejection models across all coverages. For this, we introduce the Area under the Mean Additional Error curve (AMAE) metric. This metric is found in two steps: first, we calculate the mean additional error at all coverages to produce a curve like the one shown in Figure 6. Next, we calculate the area under this curve. For a test set where the samples are ordered by the coverage where they are first accepted, this can be calculated empirically using the equation:

$$
\text{AMAE} = \frac{1}{|\mathcal{D}|} \sum_{i=1}^{|\mathcal{D}|} \sum_{j=1}^{i} \frac{\Delta E(x^i, s^i_c, s^i_{gs}, y^i|f, \ell)}{i}.
$$

The AMAE can then be used to directly compare rejection models across all target coverages. For all proposed metrics (AE, MAE, AMAE), a lower value corresponds to a better performance.

3.3. DAER

We approach the task of seed rejection by using a re-gressed estimate of the additional error (Equation 1) as a scoring function to which a threshold can be applied. This regression is learned through a novel method we call Dual-loss Additional Error Regression (DAER). Core to DAER is the separation of the additional error regression into two components corresponding to the challenges described in the introduction. The correctness loss, which addresses the subgoal understanding the cause of error, is a classifier which estimates the likelihood that seed is correct. The regression loss, which addresses the subgoal understanding task model response, estimates the additional error given that the seed is incorrect. That is, the regression loss is only used for training when the given seed is incorrect. This overall procedure is shown in Figure 4.

Mathematically, the correctness and regression outputs can be used to calculate the expected additional error:

$$
\mathbb{E}(\Delta E(x^i, s^i_c, s^i_{gs}, y^i|f, \ell)) = p(\text{seed_correct})\mathbb{E}(\Delta E|\text{seed_correct}) + p(\neg\text{seed_correct})\mathbb{E}(\Delta E|\neg\text{seed_correct}).
$$

Since the additional error for a correct seed is always zero,
this simplifies to:

\[ E(\mathbf{AE}(x^i, s^i_c, s^i_p, y^i_t, \ell)) = p(\neg \text{seed correct}) E(\mathbf{AE} | \neg \text{seed correct}) \].

We use this formula to predict the additional error at inference time, but not during training. Instead, we train \( p(\neg \text{seed correct}) \) and \( E(\mathbf{AE} | \neg \text{seed correct}) \) with separate losses, a method that is the key component of DAER. While DAER’s training method is mathematically equivalent to regressing the additional error directly, we show in Section 4.3 that separating the two components significantly improves performance.

4. Experiments

Our seed rejection method is applicable to a wide variety of problems, as it is fully specified by a four-tuple containing a (fixed) task model, a rejection model architecture, a performance measure, and a definition of a correct seed. In this section, we demonstrate this flexibility by showing state-of-the-art performance on two disparate tasks: keypoint-conditioned viewpoint estimation and hierarchical scene classification. Extra details on training and evaluation for both tasks are available in our supplementary material and code repository.

4.1. Keypoint-Conditioned Viewpoint Estimation

Keypoint-conditioned viewpoint estimation [52] is a human-in-the-loop extension of the canonical computer-vision task of viewpoint estimation [54, 50, 68, 35, 32]. In this task, a human annotator is given an image of a vehicle, and asked to click a keypoint such as “front right tire.” This human-produced information is then combined with features from a convolutional neural network to estimate the camera viewpoint more accurately than would be possible without the keypoint [50, 54].

In this work, we use the Click-Here CNN architecture [52] as our task model and, with modified output layers, our rejection model. For evaluation, our performance measure is the geodesic on the unit sphere, following convention [52, 50, 54]. However, it is impractical to use this measure during training due to the computational difficulty of calculating the matrix logarithm. Instead, our rejection model predicts rotational displacement in terms of Larochelle et al.’s distance [29],

\[ d = ||I - A_2 A_1^T||_F \],

where \( A_1 \) and \( A_2 \) are the rotation matrices produced by the ground-truth and regressed Euler angles.

While it is intuitive to define a correct seed as one that exactly matches the gold-standard seed, the Click-Here CNN architecture uses a 46x46 one-hot grid as a seed, which makes it unlikely that a randomly selected point will match the gold-standard keypoint. Therefore, defining a correct seed in this way would result in a rejection model whose objective effectively reduces to regressing the additional error directly. Instead, we define a correct seed as a seed for which the additional error is zero:

\[ p(\text{seed correct}) = \begin{cases} 0 & \text{AE} = 0 \\ 1 & \text{AE} \neq 0 \end{cases} \].

In addition to more effectively balancing correct and incorrect seeds, defining a correct seed in this way encourages the rejection model to take a shortcut by learning the interaction between the task model and primary input prior to considering the seed. For example, the left and center cases in Figure 2 can be accepted without considering the location of the seed.

Training During training, candidate seeds are generated by randomly sampling a pixel within the input image crop. For the correctness loss, we use binary cross-entropy, while we follow the common convention of using binned cross-entropy for the regression loss [54, 50].

Evaluation We maintain the human-in-the-loop motivation of the original work by evaluating with crowdsourced keypoints for our seeds. We collected a total of 6,042 keypoints on the PASCAL3D+ validation set [62] from US-based annotators via Amazon Mechanical Turk. In order to produce a representative seed distribution for validation, we divide the PASCAL3D+ validation set and corresponding crowdsourced seeds into five folds such that no vehicle crop appears in more than one fold, and report the mean across folds.

Baselines Our baselines for seed rejection on the keypoint-conditioned viewpoint estimation task are:

- Softmax Response (S.R.): The largest value of the softmax output. This was shown by Geifman & El-Yaniv [13] to perform best on selective prediction, the task most similar to seed rejection.
- Known Distance: Oracle knowledge of the candidate seed’s Euclidian distance from the gold-standard seed. This has a relation to crowdsourcing approaches, which seek to minimize error in the input space.
- Task Network Entropy: The distributional entropy of the output of the task model.
- Task Network Percentile: 10,000 samples are taken from the task model’s output distribution, and the 80th percentile difference between all samples and the mean is used as our rejection criteria. Results for other percentiles are given in supplementary material.

Results We show in Table 1 that DAER outperforms baselines on the keypoint-conditioned viewpoint estimation task. We highlight specific examples in Figures 5 and 7. In 5-(A), we see an extreme case where the gold-standard is

1https://github.com/lemmersj/ground-truth-or-daer
near the decision boundary and there is a high additional error even though the candidate seed is near the gold-standard seed. This causes the known distance baseline to fail by accepting the candidate seed early, while DAER and baselines based on the task model’s output recognize a high probability of error and accept this candidate seed late. In 5-(B), we highlight a case where DAER successfully recognizes that while the geodesic error for the candidate seed is high, the ground-truth seed will not provide an improved estimate of the camera viewpoint. 5-(C) represents a similar case in which the gold-standard seed causes error in the output, but in this case the candidate seed produces a better output, despite a mismatch between the keypoint label and location. In 5-(D), we see a failure case, where DAER is unable to accurately estimate the task model’s decision boundary, resulting in early acceptance of a poor seed.

4.2. Hierarchical Scene Classification

Hierarchical scene classification [21, 27, 57] is an extension of fine-grained classification [56, 67, 31, 63] in which information about the coarse scene categorization—such as “indoor”—is given to a classifier alongside the image to help determine the fine-grained scene categorization—such as “ballroom”—of an image. In this work, we train and evaluate on the SUN397 dataset [63], a dataset of over 130,000 images across 397 classes, and use the Plugin Network architecture developed by Koperski et al. [27] as our task model. For this problem, we define the correct seed as the seed that matches the gold-standard coarse classification. The performance measure is given as:

$$\ell(f(x, s), y) = \begin{cases} 0 & f(x, s_c) = y \\ 100 & f(x, s_c) \neq y \end{cases} \quad (8)$$

With this performance measure, the MAE corresponds to the percent difference in accuracy caused by using candidate seeds in place of the gold-standard seeds at a given coverage.

Training The rejection model, based on a pretrained ResNet-18 architecture, is trained using a randomly selected coarse category as the seed, and validation is performed using all potential seeds for a primary input. The instance of the rejection model with the lowest validation MAE is used for testing. Full training details are available in the supplementary material.

Evaluation For the hierarchical scene classification task, the seed is produced via a classification model trained to predict one of the 7 coarse category combinations (details in supplementary). We train five rejection models and five seeding models. This allows us to calculate the standard error across 5 runs for the baseline methods, and across 25 runs for the learned rejection models.

Baselines As our baselines, we use the task network entropy and softmax response scores described in Section 4.1. Since the seed is provided by a DNN classifier, we apply these baselines to both the output of the task model, which we prefix with the term “fine”, and the output of the seeding model, which we prefix with the term “coarse”.

Results We see in Table 2 that DAER significantly outperforms baselines on the seed rejection task for hierarchical scene classification under the aggregate MAE metric. Further, we see in Figure 6, that DAER outperforms all baselines on the MAE metric at every coverage greater than 0.197, which corresponds to all cases where fewer than 80.3% of seeds are rejected. At this crossover point, the MAE is approximately 0.45, meaning in about 1 out of every 222 samples an incorrect answer will be caused by an incorrect seed.

We also consider the goal of minimizing the number of rejected seeds for a target MAE under the assumption of oracle thresholding. We consider the cases where it is ac-
Regression: The way DAER combines its outputs during the task model response, which correspond to correctness and regression losses, respectively, in DAER. While we have shown that DAER outperforms the baselines, we have not yet examined the contributions of each subgoal. To do this, we perform three ablations:

1. Correctness: It may be adequate to guess whether or not the seed is a cause of error. To test this, we use the correctness loss alone as the rejection criteria.

2. Regression: The way DAER combines its outputs during the functionality of DAER: first, in both tasks the correctness loss outperforms the regression loss. Second, even without the seed, understanding the task model’s response to the given primary input and seed.

3. No Seed: While we, in some cases, encourage simplifying the goal of understanding the task model’s response by learning which primary inputs are difficult, we would like to ensure that the model does not rely solely on this shortcut. To test if this is the case, we regress the additional error without access to the seed.

We see the results of these ablations in Table 4, which reveals two interesting phenomena that provide insight into the functionality of DAER: first, in both tasks the correctness loss outperforms the regression loss. Second, even without the seed, understanding the task model’s response to the primary input is competitive with some baselines.

The fact that the correctness loss outperforms the regression loss suggests that classifying seeds as correct and incorrect—understanding the cause of error—is the easier task, and that this rough categorization combined with its implicit confidence is a moderately effective rejection method. However, the fact that it is improved by a conditioned version of regressing the additional error shows us both that eliminating cases where the seed is correct results in an easier regression problem, and that a rejection model trained to solve this regression problem can learn to estimate the task model’s response.

Further, the fact that performance of a rejection model trained without access to the seed is comparable to baselines on both tasks suggests that it is possible, but not optimal, to perform seed rejection based on the sensitivity of a primary input and task model to the seed. We see why this might be the case in Figure 7, where the most accurate seed rejection can be performed by regressing the additional error, but rejecting an unknown keypoint on the rightmost im-

### Table 2: AMAE for baselines and DAER on the HSC task (lower is better).

| Method                  | AMAE          |
|------------------------|---------------|
| Random                 | 6.17 ± 1.1e^{-1} |
| Fine Softmax Response  | 3.35 ± 4.1e^{-2} |
| Fine Entropy           | 3.29 ± 3.8e^{-2} |
| Coarse Softmax Response| 1.75 ± 4.6e^{-2} |
| Coarse Entropy         | 1.75 ± 4.6e^{-2} |
| DAER                   | 1.62 ± 3.4e^{-3} |

### Table 3: The percentage of seeds which must be rejected for various target MAEs on the hierarchical scene classification task (lower is better), as well as the percent reduction from using DAER over the next-best baseline.

| Method                  | Target MAE | Relative Reduction² |
|------------------------|------------|---------------------|
| Fine Softmax Response  | 1          | 85.0%               |
| Fine Entropy           | 2.5        | 64.3%               |
| Coarse Softmax Response| 5          | 23.8%               |
| Coarse Entropy         | 1          | 51.8%               |
| DAER                   | 2.5        | 85.0%               |
| DAER                   | 5          | 0.6                 |

²Calculated: \( \frac{\text{Coarse Entropy} - \text{DAER}}{\text{Coarse Entropy}} \)

Figure 6: The mean additional error compared to the proportion of seeds accepted (coverage) for the hierarchical scene classification task (lower is better). The dark lines represent the mean of all runs. The shaded area represents one standard error.
Table 4: AMAE for DAER and its individual subgoals (lower is better).

|            | KCVE          | HSC          |
|------------|---------------|--------------|
| Correctness| 0.2937 ± 1.79 | 1.79 ± 2.3e⁻²|
| Regression | 1.1633 ± 2.05 | 2.05 ± 1.1e⁻²|
| No Seed    | 0.8002 ± 2.28 | 2.28 ± 2.1e⁻²|
| DAER       | 0.2864 ± 1.62 | 1.62 ± 3.4e⁻³|

The goal of seed rejection is to reduce the potential impact of incorrect seeds at inference time. In doing so, it reduces the number of seeds required for a target accuracy, thereby lowering the cost of deployment and making such artificial intelligence solutions more broadly accessible. Since the ultimate goal of seed rejection is to correct inferences that are already incorrect for a fixed task model, it is unable to increase the impact of any bias or failure mode over a deployment that does not utilize a rejection model although, like all models, a DAER rejection model is subject to its own failure modes.

One notable exception to this is if DAER is extended to dataset curation, either through active learning [65] or by removing training data that degrades model performance [46]. Since such an application would establish a bidirectional dependency between the task and rejection models (i.e., the task model trains the rejection model which trains the task model), the ultimate point of convergence is unclear, and may amplify biases or blind spots. As such, we do not recommend a direct extension of the findings of DAER to dataset curation without a thorough investigation of this phenomenon.

6. Conclusion

In this work, we introduced the novel problem of seed rejection, addressing for the first time the impact of individual incorrect seeds on a model’s performance. In problem-agnostic terms, we introduce the evaluation metrics of additional error (AE), mean additional error (MAE), and area under the mean additional error curve (AMAE), and designate two meaningful subgoals: understanding the cause of error, and understanding the task model response. These subgoals motivate the Dual-loss Additional Error Regression (DAER) method, which we show can reduce the number of required re-annotations for a target MAE by over 23% compared to the best-performing baseline.

Acknowledgements Toy...
References

[1] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual Question Answering. In 2015 IEEE International Conference on Computer Vision (ICCV), pages 2425–2433, Santiago, Chile, Dec. 2015. IEEE. 1, 3, 8

[2] Mohamed El Banani and Jason J. Corso. Adviser Networks: Learning What Question to Ask for Human-In-The-Loop Viewpoint Estimation. arXiv:1802.01666 [cs], Oct. 2018. arXiv: 1802.01666. 3

[3] Luca Bertinetto, Jack Valmadre, João F. Henriques, Andrea Vedaldi, and Philip H. S. Torr. Fully-Convolutional Siamese Networks for Object Tracking. In European conference on computer vision, pages 850–865, Amsterdam, The Netherlands, June 2016. Springer International Publishing. 8

[4] Steve Branson, Catherine Wah, Florian Schroff, Boris Babenko, Peter Welinder, Pietro Perona, and Serge Belongie. Visual Recognition with Humans in the Loop. In European Conference on Computer Vision, volume 6314, pages 438–451, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. 3

[5] S. Caelles, K. K. Maninis, J. Pont-Tuset, L. Leal-Taixe, D. Cremers, and L. Van Gool. One-Shot Video Object Segmentation. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 5320–5329, Honolulu, HI, July 2017. IEEE. 8

[6] C. Chow. On optimum recognition error and reject tradeoff. IEEE Transactions on Information Theory, 16(1):41–46, Jan. 1970. 2, 3

[7] Peng Dai, Christopher H. Lin, Mausam, and Daniel S. Weld. POMDP-based control of workflows for crowdsourcing. Artificial Intelligence, 202:52–85, Sept. 2013. 1, 2

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In 2009 IEEE conference on computer vision and pattern recognition, pages 248–255, Miami, FL, 2009. IEEE. 1, 2

[9] Samuel Dodge and Lina Karam. Understanding How Image Quality Affects Deep Neural Networks. In Proceedings of the 8th International Conference on Quality of Multimedia Experience, pages 1–6, Lisbon, Portugal, June 2016. Institute of Electrical and Electronics Engineers Inc. 1

[10] Kevin Eykholt, Ivan Evtimov, Earlene Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust Physical-World Attacks on Deep Learning Visual Classification. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 1625–1634, Salt Lake City, UT, USA, June 2018. IEEE. 1

[11] Heng Fan and Haibin Ling. Siamese Cascaded Region Proposal Networks for Real-Time Visual Tracking. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 7944–7953, Long Beach, CA, USA, June 2019. IEEE. 8

[12] Giorgio Fumera and Fabio Roli. Support Vector Machines with Embedded Reject Option. In Gerhard Goos, Juris Hartmanis, Jan van Leeuwen, Seong-Whan Lee, and Alessandro Verri, editors, Pattern Recognition with Support Vector Machines, volume 2388, pages 68–82. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002. 3

[13] Yonatan Geifman and Ran El-Yaniv. Selective Classification for Deep Neural Networks. In Advances in Neural Information Processing Systems, pages 4878–4887, Long Beach, CA, USA, 2017. Curran Associates. 3, 4, 5

[14] Yonatan Geifman and Ran El-Yaniv. SelectiveNet: A Deep Neural Network with an Integrated Reject Option. In Proceedings of the 36th international conference on Machine learning, pages 2151–2159, Long Beach, CA, USA, 2019. ACM Press. 1, 2, 3

[15] Yonatan Geifman, Guy Uziel, and Ran El-Yaniv. Bias-Reduced Uncertainty Estimation for Deep Neural Classifiers. In Proceedings of the Seventh International Conference on Learning Representations, pages 1–14, New Orleans, LA, 2019. 2, 3

[16] Brent Griffin, Victoria Florence, and Jason J. Corso. Video Object Segmentation-based Visual Servo Control and Object Depth Estimation on a Mobile Robot Platform. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision, pages 1647–1657, Snowmass Village, CO, 2020. IEEE. 3

[17] Brent A. Griffin and Jason J. Corso. BubbleNets: Learning to Select the Guidance Frame in Video Object Segmentation by Deep Sorting Frames. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 8914–8923, 2019. 3

[18] Danna Gurari, Suyog Dutt Jain, Margrit Betke, and Kristen Grauman. Pull the Plug? Predicting If Computers or Humans Should Segment Images. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 382–391, Las Vegas, NV, USA, June 2016. IEEE. 3

[19] Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P. Bigham. VizWiz Grand Challenge: Answering Visual Questions from Blind People. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3608–3617, Salt Lake City, UT, USA, June 2018. IEEE. 3, 8

[20] M. E. Hellman. The Nearest Neighbor Classification Rule with a Reject Option. IEEE Transactions on Systems Science and Cybernetics, 6(3):179–185, July 1970. 3

[21] Hexiang Hu, Guang-Tong Zhou, Zhiwei Guo, Zicheng Liao, and Greg Mori. Learning Structured Inference Neural Networks with Label Relations. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2960–2968, Las Vegas, NV, USA, June 2016. IEEE. 6

[22] Ronghang Hu, Marcus Rohrbach, and Trevor Darrell. Segmentation from Natural Language Expressions. In Proceedings of the 2016 European Conference on Computer Vision, pages 108–124, Amsterdam, The Netherlands, 2016. Springer International Publishing. 3

[23] Tianrui Hui, Si Liu, Shaofei Huang, Guanbin Li, Sansi Yu, Fuxi Zhang, and Jizhong Han. Linguistic Structure Guided Context Modeling for Referring Image Segmentation. In Proceedings of the 2020 European Conference on Computer Vision, volume 12355, pages 59–75, Virtual, 2020. Springer
Panagiotis G. Ipeirotis, Foster Provost, and Jing Wang. Quality management on Amazon Mechanical Turk. In Proceedings of the ACM SIGKDD Workshop on Human Computation - HCOMP ’10, pages 64–67, Washington DC, 2010. ACM Press.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. ReferItGame: Referring to Objects in Photographs of Natural Scenes. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 787–798, Doha, Qatar, 2014. Association for Computational Linguistics.

Anna Khoreva, Federico Perazzi, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung. Learning Video Object Segmentation from Static Images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2663–2672, Honolulu, Hawaii, USA, 2017. IEEE.

Michal Koperski, Tomasz Konopczynski, Rafal Nowak, Piotr Semberecki, and Tomasz Trzciński. Plugin Networks for Inference under Partial Evidence. In Proceedings of The 2020 IEEE Winter Conference on Applications of Computer Vision, pages 2883–2891, Snowmass Village, CO, 2020. IEEE. 1, 2, 3, 6.

Matej Kristan, Jiri Matas, Aleš Leonardis, Tomáš Vojíř, Roman Pflugfelder, Gustavo Fernández, Georg Nebeyah, Fatih Porikli, and Luka Čehovin. A Novel Performance Evaluation Methodology for Single-Target Trackers. IEEE Transactions on Pattern Analysis and Machine Intelligence, 38(11):2137–2155, Nov. 2016.

Pierre M. Larochelle, Andrew P. Murray, and Jorge Angeles. A Distance Metric for Finite Sets of Rigid-Body Displacements via the Polar Decomposition. Journal of Mechanical Design, 129(8):883–886, Aug. 2007.

Stephan J. Lemmer, Jean Y. Song, and Jason J. Corso. Crowdsourcing More Effective Initializations for Single-Target Trackers through Automatic Re-querying. In Proceedings of the 2021 Conference on Human Factors in Computing Systems, Virtual (Originally Yokohama, Japan), May 2021. ACM Press.

Z. Li and D. Hoiem. Learning without Forgetting. IEEE Transactions on Pattern Analysis and Machine Intelligence, 40(12):2935–2947, Dec. 2018.

Shuai Liao, Efstratios Gavves, and Cees G. M. Snoek. Spherical Regression: Learning Viewpoints, Surface Normals and 3D Rotations on N-Spheres. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 9751–9759, Long Beach, CA, USA, June 2019. IEEE. 2, 5.

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 Proceedings of the 2014 European Conference on Computer Vision, pages 740–755, Zurich, Switzerland, 2014. Springer International Publishing.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan Yuille, and Kevin Murphy. Generation and Comprehension of Unambiguous Object Descriptions. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 11–20, Las Vegas, NV, USA, June 2016. IEEE.

Siva Karthik Mustikovela, Varun Jampani, Shalini De Mello, Sifei Liu, Umar Iqbal, Carsten Rother, and Jan Kautz. Self-Supervised Viewpoint Learning From Image Collections. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 3970–3980, Seattle, WA, USA, June 2020. IEEE. 2, 5.

F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A Benchmark Dataset and Evaluation Methodology for Video Object Segmentation. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 724–732, Las Vegas, NV, June 2016. IEEE. 1, 3, 4, 8.

Prakruthi Prabhakar, Nitish Kulkarni, and Linghao Zhang. Question Relevance in Visual Question Answering. arXiv:1807.08435 [cs], July 2018. arXiv: 1807.08435.

Maithra Raghu, Katy Blumer, Rory Sayres, Ziad Obermeyer, Robert Kleinberg, Sendhil Mullainathan, and Jon Kleinberg. Direct Uncertainty Prediction for Medical Second Opinions. In Proceedings of the 2019 International Conference on Machine Learning, pages 5281–5290, Long Beach, CA, 2019. ACM Press.

Vikas C Raykar and Shipeng Yu. Eliminating Spammers and Ranking Annotators for Crowdsourced Labeling Tasks. Journal of Machine Learning Research, 13:491–518, Feb. 2012.

Esteban Real, Jonathon Shlens, Stefano Mazzocchi, Xin Pan, and Vincent Vanhoucke. YouTube-BoundingBoxes: A Large High-Precision Human-Annotated Data Set for Object Detection in Video. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 7464–7473, Honolulu, HI, July 2017. IEEE. 1, 2.

N Dinesh Reddy, Minh Vo, and Srinivasa G. Narasimhan. CarFusion: Combining Point Tracking and Part Detection in Video for Dynamic 3D Reconstruction of Vehicles. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 1906–1915, Salt Lake City, UT, USA, June 2018. IEEE.

Amir Rosenfeld, Richard Zemel, and John K. Tsotsos. The Elephant in the Room. arXiv:1808.03305 [cs], Aug. 2018. arXiv: 1808.03305.

Olga Russakovsky, Li-Jia Li, and Li Fei-Fei. Best of both worlds: Human-machine collaboration for object annotation. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2121–2131, Boston, MA, USA, June 2015. IEEE. 1, 2.

Jeffrey M. Rzeszotarski and Aniket Kulkarni. Instrumenting the crowd: using implicit behavioral measures to predict task performance. In Proceedings of the 24th annual ACM symposium on User interface software and technology - UIST ’11, pages 13–22, Santa Barbara, California, USA, 2011. ACM Press.

Meet Shah, Xinlei Chen, Marcus Rohrbach, and Devi Parikh. Cycle-Consistency for Robust Visual Question Answering.
[69] Yiren Zhou, Sibo Song, and Ngai-Man Cheung. On classification of distorted images with deep convolutional neural networks. In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 1213–1217, New Orleans, LA, Mar. 2017. IEEE.