ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-6874-1/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICASSP49660.2025.10888541

Robust Multi-task Adversarial Attacks Using Min-max Optimization

Jiacheng Guo
Department of CS
Cleveland State University
Cleveland, USA
j.guo58@vikes.csuohio.edu

Lei Li
Department of CS
Cleveland State University
Cleveland, USA
l.li15@vikes.csuohio.edu

Haochen Yang
Department of CS
Cleveland State University
Cleveland, USA
h.yang15@vikes.csuohio.edu

Baocheng Geng
Department of CS
University of Alabama at Birmingham
Birmingham, USA
bgeng@uab.edu

Hongkai Yu
Department of ECE
Cleveland State University
Cleveland, USA
h.yu19@csuohio.edu

Minghai Qin
Western Digital Research
Milpitas, USA
qinminghai@gmail.com

Tianyun Zhang
Department of CS
Cleveland State University
Cleveland, USA
t.zhang85@csuohio.edu

***Abstract*—Deep neural networks have achieved exceptional performance across a wide range of applications but remain susceptible to adversarial attacks. While most prior research has focused on single-task scenarios, increasing attention is being directed toward adversarial attacks targeting multiple tasks simultaneously. However, existing methods often fail to balance attack performance across tasks in a multi-task model. These approaches typically aim to maximize the model's overall loss, neglecting task-specific attack difficulties, which results in imbalanced attack performance among tasks. To address this challenge, we propose a novel multi-task adversarial attack method that ensures robust and balanced attack performance across multiple tasks. Our approach dynamically updates task-specific weighting factors through a min-max optimization during the attack, optimizing the worst-case attack performance across all tasks. Experimental results demonstrate that our method significantly enhances the worst-case attack performance across diverse datasets and attack strategies compared to existing approaches. By dynamically adjusting the attack intensity on the least vulnerable tasks, the min-max optimization significantly improves overall attack effectiveness as well as the worst-case performance by balancing the task weights.**

***Index Terms*—multi-task learning, adversarial attacks, deep learning.**

I. Introduction

Deep Neural Networks (DNNs) have achieved remarkable performance across a wide range of applications, as demonstrated in various studies [1]–[3]. Despite their successes, DNNs remain vulnerable to adversarial attacks [4], [5]. These attacks involve adding subtle perturbations to input data, which can cause the network to produce incorrect outputs [4], [6]. Often, these perturbations are small and imperceptible to the human eye, yet they can significantly disrupt the model's performance [7], [8]. While much of the existing research on adversarial attacks focuses on single-task scenarios, there is a growing interest in attacks that target multiple tasks simultaneously.

Corresponding author: Tianyun Zhang (t.zhang85@csuohio.edu).

Multi-task learning (MTL) is a subfield of machine learning where multiple related tasks are learned concurrently [9]–[11]. In MTL, a single model with shared parameters is trained instead of training separate models for each task. This approach improves model efficiency and generalization while reducing computational and storage costs [12]–[14].

However, the shared parameters in MTL also create a unique opportunity for adversarial attacks. An attacker could target multiple tasks simultaneously within the shared model, which could be particularly effective in systems that rely on several interdependent tasks, such as autonomous driving. These systems perform various tasks, including object detection, depth estimation, and normal detection. An attack on multiple components of such systems could lead to significant disruption. Multi-task adversarial attacks aim to generate subtle, often imperceptible perturbations that degrade the performance of all tasks effectively [8], [15].

A key challenge in existing multi-task adversarial attack methods is balancing the attack effectiveness across different tasks. In MTL frameworks, some tasks may be easier to attack than others, leading to uneven performance degradation. This imbalance results in an unsatisfactory overall attack effectiveness, where some tasks are significantly impacted while others are only marginally affected. Therefore, developing an advanced attack method that balances the effectiveness across tasks is crucial for ensuring a uniformly effective attack.

In this paper, we propose a novel multi-task adversarial attack method to achieve robust attack performance across different tasks. The min-max optimization is applied in our proposed method, it can dynamically adjust the weighting factors of each task to optimize the worst-case attack performance across all tasks. Experimental results demonstrate that our proposed method consistently improves the worst-case attack performance on different datasets with different attack strategies compared with prior methods.

Our main contributions in this paper are as follows:

- We propose a novel multi-task adversarial attack method to achieve robust attack performance across different tasks. In our proposed method, the weighting factors corresponding to different tasks are updated dynamically during the adversarial attack, this can effectively optimize the worst-case attack performance.
- We evaluate our proposed method on NYUv2 [16] and Cityscapes [17] datasets with L_2 and L_∞ attack strategies, based on the projected gradient descent (PGD) algorithm [18]. Experimental results demonstrate that our proposed method consistently improves the worst-case attack performance compared with prior methods.

II. Related Work

A. Multi-task Learning

Multi-task learning (MTL) is a machine learning paradigm that trains multiple related tasks simultaneously, offering benefits like improved data efficiency, reducing overfitting, and enhancing generalization effectiveness [9]–[11]. It has gained traction in specific fields such as natural language processing and computer vision [19], [20]. MTL techniques are often categorized into areas like regularization, pre-training, relationship learning, feature propagation, and optimization [2], [21]–[23]. Approaches to MTL can be divided into joint training and multi-step training methods based on task relationships [8], [19]. The field has progressed from traditional methods to deep learning and pre-trained models, with recent innovations focusing on task-promotable, task-agnostic training, and zero-shot learning [20], [24]. Additionally, MTL's application in distributed and streaming contexts underscores their growing versatilities [25].

B. Adversarial Attacks

Adversarial attacks pose a significant challenge for DNNs by exploiting tiny perturbations that can drastically alter model predictions [26], [27]. These perturbations, often imperceptible to humans, expose critical weaknesses in DNN architectures [4], [28], [29]. One influential method for generating adversarial attacks is gradient optimization. [15] introduced universal adversarial perturbations, which manipulate the gradient of the loss function to fool classifiers. This work paved the way for many gradient-based attack methods. In MTL, adversarial attacks pose unique challenges due to the need to balance multiple tasks. [30] addressed gradient imbalance in MTL with GradNorm, a technique that normalizes gradients to ensure balanced loss optimization across tasks, indirectly enhancing robustness. [7] expanded adversarial attacks to MTL, showing that multiple tasks could be attacked simultaneously. Recently, a stealthy attack in MTL is proposed by [31] where the targeted task is significantly attacked while non-targeted tasks still preserve their original performance. [8], [32] further explored the complexities of achieving robustness in MTL, highlighting the intricate interactions between tasks. Different from [8], [32] that applies equal weighting factors to attack different tasks, we propose to update the weighting factors dynamically in multi-task adversarial attacks.

III. Methodology

A. Problem Formulation

Assume we have a dataset with input data x, and the ground-truth $y = (y_1, y_2, ..., y_i, ..., y_k)$, where y_i denotes the ground-truth for the i-th task. Also, we have a pretrained multi-task model for this dataset, with the loss function $L_i(x, y_i)$ corresponding to the i-th task. The multi-task adversarial attack problem is given by

$$\underset{||\delta||_p \leq \epsilon, x+\delta \in \mathcal{B}}{\text{maximize}} \quad \sum_{i=1}^{k} w_i L_i(x+\delta, y_i), \tag{1}$$

where δ is the adversarial noise, ϵ constrains the strength of the adversarial noise, and $\mathcal{B}$ is the box constraint to ensure that the adversarial example is valid, w_i denotes the weighting factor corresponding to the i-th task, and k denotes the total number of tasks. In prior works [8], [32], equal weighting factors are given to all tasks, in which $w_i = 1/k$ for all i. Another approach that may mitigate the imbalanced attack performance on each task is normalization, in which $w_i = 1/L_i(x, y_i)$.

Different from the above approaches, we propose to formulate multi-task adversarial attacks as a robust optimization problem, which is given by

$$\underset{||\delta||_p \leq \epsilon, x+\delta \in \mathcal{B}}{\text{maximize}} \ \underset{w \in P}{\text{minimize}} \quad \sum_{i=1}^{k} w_i L_i(x+\delta, y_i) + \frac{\gamma}{2}\|w - l/k\|_2^2, \tag{2}$$

where w denotes the collection of w_i, P denotes the probability simplex $P = \{w \,|\, l^T w = 1, w_i \in [0,1], \forall i\}$, l denotes an all-ones vector with the same size as w, and γ is a regularization parameter. Here, the weighting factors w_i are the optimization variables that target on balancing the loss of different tasks. The reason to include a regularization term in this problem is to avoid deriving a one-hot vector for w and thus it improves the generalizability to different tasks.

B. Problem Solving Algorithm

We apply the alternating projected gradient descent-ascent (APGDA) method [33] to solve problem (2). Specifically, we solve the outer maximization problem using projected gradient ascent to generate adversarial attacks and solve the inner minimization program using projected gradient descent to adjust the weighting factors corresponding to different tasks. The details of the solving algorithm are summarized in Algorithm 1.

In Algorithm 1, when w is fixed, the update of δ is given by

$$\delta^{(t)} = \text{proj}_{\mathcal{X}}\Big(\delta^{(t-1)} + \alpha_1 \nabla_\delta \Big(\sum_{i=1}^{k} w_i L_i(x+\delta^{(t-1)}, y_i)\Big)\Big), \tag{3}$$

where α_1 is the learning rate to update δ, $\text{proj}_{\mathcal{X}}(\cdot)$ denotes the Euclidean projection onto the set $\mathcal{X} = \{\delta \,|\, ||\delta||_p \leq \epsilon, x+\delta \in \mathcal{B}\}$. The update of δ can be derived based on the prior PGD adversarial attack [18].

Algorithm 1 APGDA method to solve problem (2)

1: Input: input data x, and the ground-truth $y = (y_1, y_2, ..., y_i, ..., y_k)$, pretrained multi-task model and attack steps T, $w^{(0)} = l/k$.
2: **for** $t = 1, 2, \ldots, T$ **do**
3: *outer maximization*: fixing $w = w^{(t-1)}$, update adversarial noise $\delta^{(t)}$ with projected gradient ascent.
4: *inner minimization*: fixing $\delta = \delta^{(t)}$, update $w^{(t)}$ with projected gradient descent
5: **end for**

For the inner minimization problem, w is updated by

$$w^{(t)} = \text{proj}_P\Big(w^{(t-1)} - \alpha_2 \nabla_w f(w^{(t-1)})\Big), \tag{4}$$

where $f(w) = \sum_{i=1}^{k} w_i L_i(x+\delta, y_i) + \frac{\gamma}{2}\|w - l/k\|_2^2$, α_2 is the learning rate to update w, and $\text{proj}_P(\cdot)$ denotes the Euclidean projection onto the simplex set P, the closed-form solution of this kind of Euclidean projection is derived by [34].

IV. Experiments

A. Experimental Settings

To evaluate the effectiveness of our proposed robust multi-task adversarial attack method, we implement the experiments on two multi-domain datasets: NYUv2 [16] and Cityscapes [17].

By default, three tasks are trained on the NYUv2 dataset: 13-class semantic segmentation, depth estimation, and surface normal prediction. In the Cityscapes dataset, we also train three tasks respectively: 19-class semantic segmentation, disparity estimation (i.e., inverse depth estimation), and a newly proposed 10-class part segmentation following [35]. In the multi-task adversarial attacks, we compare the performance of the prior methods with the proposed min-max optimization method. All the basic settings in the experiments follow the multi-task attention network proposed by [36] which is based on ResNet-50 [37].

B. Evaluation Metrics

In the NYUv2 dataset, three tasks are evaluated via mean intersection over union (mIoU), absolute error (aErr), and mean angle distances (mDist). For the Cityscapes dataset, both the semantic segmentation and part segmentation tasks are evaluated by mean intersection over union (mIoU), and the disparity estimation task is evaluated by absolute error (aErr) followed by [38]. We also report the overall multi-task attack performance $\triangle_{MTL}$ following [39] by

$$\triangle_{MTL} = \frac{1}{k}\sum_{i=1}^{k} \triangle_i, \tag{5}$$

where i denotes each task, and k denotes the total number of tasks. The attack effect of each task (denoted by $\triangle_i$) is calculated by

$$\triangle_i = \begin{cases} \frac{M_{m,i} - M_{b,i}}{M_{b,i}} & \text{, if } M_{m,i} \geq M_{b,i} \\ \frac{M_{b,i} - M_{m,i}}{M_{m,i}} & \text{, otherwise,} \end{cases} \tag{6}$$

where M denotes the performance after attack, m denotes each task, and b is the baseline performance for each task. If a higher value means better effectiveness in the corresponding metric, we calculate the difference between the attacked performance and baseline, then divided by the baseline performance. Similarly, if a lower value means better effectiveness in the corresponding metric, the difference is calculated between baseline and attacked performance, then divided by the attacked performance.

Furthermore, we extract the performance of the most difficult task to be attacked (i.e., the worst-case performance) which is denoted by $\triangle_{Wor}$. The worst-case performance could be formulated as (7).

$$\triangle_{Wor} = \min_i \triangle_i \tag{7}$$

C. Experimental Results and Analysis

TABLE I
EXPERIMENTAL RESULTS USING PGD L_2 ATTACK METHOD ON NYUV2 DATASET. FOR EACH TASK, "↑" MEANS HIGHER BETTER AND "↓" MEANS LOWER BETTER.

Tasks Metrics	Segment [mIoU(↑)]	Depth [aErr(↓)]	Normal [mDist(↓)]	$\triangle_{MTL}$ (%)	$\triangle_{Wor}$ (%)
Baseline	46.56	40.57	23.41	0	0
ϵ=5					
Equal	18.63	84.43	34.70	-48.15	-32.52
Normalize	18.23	118.31	38.22	-55.10	-38.75
Min-max	18.71	126.39	41.33	**-57.02**	**-43.35**
ϵ=10					
Equal	12.27	104.65	40.16	-58.86	-41.71
Normalize	11.23	156.25	45.09	-66.00	-48.08
Min-max	11.86	174.39	51.48	**-68.60**	**-54.53**

TABLE II
EXPERIMENTAL RESULTS USING PGD L_2 ATTACK METHOD ON CITYSCAPES DATASET. FOR EACH TASK, "↑" MEANS HIGHER BETTER AND "↓" MEANS LOWER BETTER.

Tasks Metrics	Segment [mIoU(↑)]	Part Seg [mIoU(↑)]	Disp [aErr(↓)]	$\triangle_{MTL}$ (%)	$\triangle_{Wor}$ (%)
Baseline	54.20	51.82	81.51	0	0
ϵ=5					
Equal	24.04	29.92	264.38	**-55.69**	-42.26
Normalize	26.45	27.81	156.78	-48.51	-46.33
Min-max	26.13	27.76	220.66	-53.76	**-46.43**
ϵ=10					
Equal	19.53	20.18	333.53	-66.86	-61.05
Normalize	20.82	20.89	195.62	-59.87	-58.33
Min-max	20.35	20.10	369.39	**-67.19**	**-61.21**

We evaluate the attack effects by varying the pixel levels of input images in two scenarios: PGD L_∞ and PGD L_2 attacks. In the PGD L_∞ attack scenario, the perturbation parameter ϵ is set to $2/255$, $4/255$, $8/255$, and $16/255$, representing changes of 1, 2, 4, and 8 pixel levels, respectively. For PGD L_2 attack, we set ϵ to 5 and 10.

Tables I and II present the results of the PGD L_2 attack method on NYUv2 and Cityscapes datasets, respectively. Note that the baseline model indicates that ϵ=0 without any

adversarial attack. In the "Equal" method, the total loss is calculated as the sum of losses from three tasks which means all those three tasks are given equal weighting factors. It shows a noticeable imbalance in attack effects across the three tasks.

To address this problem, a normalization strategy is employed, which incorporates a task-specific weighting factor to each task to balance the effects among the tasks. However, the normalization method fails to achieve a consistent performance on different datasets. Generally, it performs well on the NYUv2 dataset but performs quite poorly on the Cityscapes dataset especially when ϵ=10. Comparatively, our proposed min-max optimization method consistently achieves robust attack performance across different tasks in two datasets with different values of ϵ. The min-max method highly improves the worst-case attack performance and also maintains the overall attack performance compared with other methods. Specifically, the min-max optimization method is applied to further fine-tune the weighting factors of each task automatically, aiming at a more balanced and effective attack. Both the overall as well as the worst-case attack effects could be further enhanced in both NYUv2 and Cityscapes datasets, except ϵ=5 in Cityscapes compared with the "Equal" method.

TABLE III
EXPERIMENTAL RESULTS USING PGD L_∞ ATTACK METHOD ON NYUV2 DATASET. FOR EACH TASK, "↑" MEANS HIGHER BETTER AND "↓" MEANS LOWER BETTER.

Tasks Metrics	Segment [mIoU(↑)]	Depth [aErr(↓)]	Normal [mDist(↓)]	$\triangle_{MTL}$ (%)	$\triangle_{Wor}$ (%)
Baseline	46.56	40.57	23.41	0	0
ϵ=2/255					
Equal	25.28	67.28	29.82	-35.64	-21.50
Normalize	26.21	86.09	31.81	-40.99	-26.40
Min-max	26.69	87.99	33.00	**-41.88**	**-29.06**
ϵ=4/255					
Equal	15.72	89.7	35.96	-51.97	-34.89
Normalize	15.87	123.93	39.44	-57.94	-40.64
Min-max	17.05	132.41	43.59	**-59.68**	**-46.29**
ϵ=8/255					
Equal	8.85	113.42	42.31	-63.30	-44.67
Normalize	8.45	167.49	47.35	-69.40	-50.56
Min-max	9.97	189.52	56.12	**-71.82**	**-58.29**
ϵ=16/255					
Equal	4.90	130.65	47.04	-69.55	-50.23
Normalize	4.11	205.39	52.90	-75.72	-55.74
Min-max	5.41	234.53	66.91	**-78.70**	**-65.01**

The results of multi-task adversarial attacks using PGD L_∞ attack method on NYUv2 and Cityscapes datasets are presented in Tables III and IV, respectively. Similar to the case of PGD L_2 attack, the normalization method performs well in maintaining the worst-case performance only for the NYUv2 dataset. For the Cityscapes dataset, the normalization method achieves relatively worse overall performance and worst-case performance when ϵ is $8/255$ or $16/255$. Different from the normalization method, the min-max optimization method consistently performs well on the overall performance as well as worst-case performance for those two datasets with different values of ϵ. In the NYUv2 dataset, min-max significantly improves the overall attack effect as well as worst-case attack effectiveness compared with prior methods. In the Cityscapes dataset, a noticeable enhancement occurs when using min-max optimization to improve the worst-case attack. When compared with the equal weighting method, the min-max optimization improves the worst-case attack performance by 6% on average with similar or better overall attack performance.

TABLE IV
EXPERIMENTAL RESULTS USING PGD L_∞ ATTACK METHOD ON CITYSCAPES DATASET. FOR EACH TASK, "↑" MEANS HIGHER BETTER AND "↓" MEANS LOWER BETTER.

Tasks Metrics	Segment [mIoU(↑)]	Part Seg [mIoU(↑)]	Disp [aErr(↓)]	$\triangle_{MTL}$ (%)	$\triangle_{Wor}$ (%)
Baseline	54.20	51.82	81.51	0	0
ϵ=2/255					
Equal	28.23	37.00	215.27	**-46.21**	-28.60
Normalize	29.36	32.63	142.78	-41.92	-37.03
Min-max	30.34	32.85	178.71	-45.00	**-37.12**
ϵ=4/255					
Equal	18.42	23.06	341.04	**-65.87**	-55.50
Normalize	21.03	22.96	195.13	-58.37	-55.65
Min-max	22.22	23.32	331.40	-63.14	**-56.10**
ϵ=8/255					
Equal	9.57	11.76	541.38	-81.53	-77.30
Normalize	12.45	13.96	283.27	-73.77	-71.23
Min-max	10.56	10.71	642.56	**-82.38**	**-79.33**
ϵ=16/255					
Equal	3.50	8.90	853.35	**-88.93**	-82.82
Normalize	4.79	9.33	436.21	-84.82	-81.31
Min-max	7.19	8.23	1154.19	-87.93	**-84.12**

V. CONCLUSION AND FUTURE WORK

In this paper, we present a robust multi-task adversarial attack method leveraging min-max optimization. The proposed min-max optimization dynamically adjusts task-specific weighting factors, ensuring effective optimization of the task with the worst performance. Unlike existing attack methods that exhibit imbalanced attack effectiveness across tasks, our approach achieves significantly more consistent and robust performance across multiple tasks. We evaluate our method on the NYUv2 and Cityscapes datasets using PGD L_2 and PGD L_∞ attack strategies. Experimental results demonstrate that our method consistently improves the worst-case attack performance for both datasets and across various attack strategies. Our proposed method improves the worst-case attack performance by 6% on average while maintaining similar or better overall attack performance.

In addition to adversarial attacks, existing studies have explored adversarial defense techniques to enhance the robustness of deep neural networks, with adversarial training being a notable example [4]. In our future work, we aim to investigate advanced defense strategies to further enhance adversarial robustness and balance task performance in multi-task deep neural networks.

ACKNOWLEDGMENTS

This research was supported by the National Science Foundation under award CNS-2245765.

REFERENCES

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[2] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal processing magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[3] D. Andor, C. Alberti, D. Weiss, A. Severyn, A. Presta, K. Ganchev, S. Petrov, and M. Collins, "Globally normalized transition-based neural networks," *arXiv preprint arXiv:1603.06042*, 2016.

[4] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=rJzIBfZAb

[5] D. Wang, C. Li, S. Wen, S. Nepal, and Y. Xiang, "Defending against adversarial attack towards deep neural networks via collaborative multi-task training," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 2, pp. 953–965, 2020.

[6] N. Carlini, P. Mishra, T. Vaidya, Y. Zhang, M. Sherr, C. Shields, D. Wagner, and W. Zhou, "Hidden voice commands," in *25th USENIX security symposium (USENIX security 16)*, 2016, pp. 513–530.

[7] P. Guo, Y. Xu, B. Lin, and Y. Zhang, "Multi-task adversarial attack," *arXiv preprint arXiv:2011.09824*, 2020.

[8] S. Ghamizi, M. Cordy, M. Papadakis, and Y. Le Traon, "Adversarial robustness in multi-task learning: Promises and illusions," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 1, 2022, pp. 697–705.

[9] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[10] L. Xu, J. Li, W. Lin, Y. Zhang, L. Ma, Y. Fang, and Y. Yan, "Multi-task rank learning for image quality assessment," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 9, pp. 1833–1843, 2016.

[11] M. Crawshaw, "Multi-task learning with deep neural networks: A survey," *arXiv preprint arXiv:2009.09796*, 2020.

[12] O. Sener and V. Koltun, "Multi-task learning as multi-objective optimization," *Advances in neural information processing systems*, vol. 31, 2018.

[13] Y. Zhang and Q. Yang, "A survey on multi-task learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 12, pp. 5586–5609, 2021.

[14] J. Guo, H. Sun, M. Qin, H. Yu, and T. Zhang, "A min-max optimization framework for multi-task deep neural network compression," *IEEE International Symposium on Circuits and Systems*, 2024.

[15] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1765–1773.

[16] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V 12*. Springer, 2012, pp. 746–760.

[17] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[18] Y. Chen and M. J. Wainwright, "Fast low-rank estimation by projected gradient descent: General statistical and algorithmic guarantees," *arXiv preprint arXiv:1509.03025*, 2015.

[19] Z. Zhang, W. Yu, M. Yu, Z. Guo, and M. Jiang, "A survey of multi-task learning in natural language processing: Regarding task relatedness and training methods," *arXiv preprint arXiv:2204.03508*, 2022.

[20] J. Yu, Y. Dai, X. Liu, J. Huang, Y. Shen, K. Zhang, R. Zhou, E. Adhikarla, W. Ye, Y. Liu *et al.*, "Unleashing the power of multi-task learning: A comprehensive survey spanning traditional, deep, and pretrained foundation model eras," *arXiv preprint arXiv:2404.18961*, 2024.

[21] T. Evgeniou and M. Pontil, "Regularized multi–task learning," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004, pp. 109–117.

[22] T. Zhang, S. Liu, Y. Wang, and M. Fardad, "Generation of low distortion adversarial attacks via convex programming," in *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2019, pp. 1486–1491.

[23] J. Guo, H. Sun, M. Qin, H. Yu, and T. Zhang, "A min-max optimization framework for multi-task deep neural network compression," in *2024 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2024, pp. 1–5.

[24] K. He, J. Zhang, Y. Yan, W. Xu, C. Niu, and J. Zhou, "Contrastive zero-shot learning for cross-domain slot filling with adversarial attack," in *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 1461–1467.

[25] R. Nassif, S. Vlaski, C. Richard, J. Chen, and A. H. Sayed, "Multitask learning over graphs: An approach for distributed, streaming machine learning," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 14–25, 2020.

[26] J. Lu, H. Sibai, and E. Fabry, "Adversarial examples that fool detectors," *arXiv preprint arXiv:1712.02494*, 2017.

[27] H. Sun, L. Fu, J. Li, Q. Guo, Z. Meng, T. Zhang, Y. Lin, and H. Yu, "Defense against adversarial cloud attack on remote sensing salient object detection," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 8345–8354.

[28] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 427–436.

[29] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*. Ieee, 2017, pp. 39–57.

[30] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, "Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks," in *International conference on machine learning*. PMLR, 2018, pp. 794–803.

[31] J. Guo, T. Zhang, L. Li, H. Yang, H. Yu, and M. Qin, "Stealthy multi-task adversarial attacks," *arXiv preprint arXiv:2411.17936*, 2024.

[32] C. Mao, A. Gupta, V. Nitin, B. Ray, S. Song, J. Yang, and C. Vondrick, "Multitask learning strengthens adversarial robustness," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*. Springer, 2020, pp. 158–174.

[33] J. Wang, T. Zhang, S. Liu, P.-Y. Chen, J. Xu, M. Fardad, and B. Li, "Adversarial attack generation empowered by min-max optimization," *Advances in Neural Information Processing Systems*, vol. 34, pp. 16 020–16 033, 2021.

[34] N. Parikh, S. Boyd *et al.*, "Proximal algorithms," *Foundations and Trends® in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[35] D. de Geus, P. Meletis, C. Lu, X. Wen, and G. Dubbelman, "Part-aware panoptic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 5485–5494.

[36] S. Liu, S. James, A. J. Davison, and E. Johns, "Auto-lambda: Disentangling dynamic task relationships," *Transactions on Machine Learning Research*, 2022.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[38] S. Liu, E. Johns, and A. J. Davison, "End-to-end multi-task learning with attention," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 1871–1880.

[39] K.-K. Maninis, I. Radosavovic, and I. Kokkinos, "Attentive single-tasking of multiple tasks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 1851–1860.