CROSS-ATTENTION GRADIENT TRANSPLANTATION (CAGT): MITIGATING GRADIENT CONFLICT IN MULTI-TASK DEEP LEARNING THROUGH CROSS-BLOCKWISE ATTENTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-task learning tries to use shared structure across tasks that are related by optimizing a joint objective. In practice though, gradients from individual tasks losses can cause conflict. This causes destructive interference that slows convergence and reduces performance. Previous approaches like PCGrad and CAGrad address this by projecting away or reweighting conflicting gradients at the vector level, but they do not explicitly exchange or reuse gradient components across tasks. This paper proposes Cross-Attention Gradient Transplantation (CAGT), which is a method that finds conflicting gradient components and replaces them using deterministic cosine similarity based cross-attention over gradient subblocks from other tasks. Task gradients are partitioned into subblocks at each optimization step, and for each subblock showing conflict, attention weights are computed from cosine similarities with other tasks' subblocks. The resulting weighted combination is rescaled to keep the original magnitude, and it is interpolated with the original subgradient. This produces adjusted task gradients that reduce adversarial interactions while keeping constructive signals. Experimental results show that CAGT outperforms traditional approaches such as PCGrad and CAGrad, achieving roughly 10% lower loss values on some multi task deep learning datasets, such as ROT-MNIST and CelebA.

1 INTRODUCTION

Cross-task deep learning seeks to train a single model on multiple related tasks simultaneously, using shared representations to improve data efficiency and generalization Caruana (1997). In principle, sharing parameters across tasks can help discover common structure and yield performance gains over training each task independently Baxter (2000). However, in practice, naively optimizing all tasks jointly usually leads to a difficult optimization problem. Jointly trained models sometimes perform worse on each task than separate single-task models Ruder (2017). This gap between theory and practice is partially because of the fact that the gradient signals from different tasks can interfere, which makes the multi-task loss landscape hard to navigate Yu et al. (2020).

A core source of difficulty is gradient conflict between tasks. We say two tasks have conflicting gradients when their update directions point away from each other (i.e. their cosine similarity is negative) Yu et al. (2020). These conflicts can stall learning, because one task's gradient update can partially cancel or even reverse another's; so, improving one task comes at the expense of another Liu et al. (2021a). Moreover, tasks usually produce gradients of very different magnitudes, so a large gradient can dominate the update even if it is misaligned with the other tasks' objectives Chen et al. (2018). Conflicting gradient directions and scale imbalances can prevent the optimizer from making progress on all tasks simultaneously Liu et al. (2021a); Chen et al. (2018).

In this work, we use cross-attention to directly modulate task gradients at a finer granularity than prior methods. We propose a framework called Cross-Attention Gradient Transplantation (CAGT), which learns to adaptively combine and reroute gradient signals across tasks at the level of gradient subspaces.

To better understand the benefits of finer grained gradient recomposition, we focus on three key aims: (1) Conflict reduction: quantify how cross block attention modulates task gradients to reduce negative interference and improve alignment between conflicting tasks; (2) Adaptive gradient combination: evaluate the effect of locally transplanting gradient components from one task to another, including the role of adaptive interpolation coefficients in controlling the strength of transplantation; (3) Empirical performance impact: measure whether these gradient modifications translate into improved task wise loss reduction and overall model convergence.

2 Related Works

Multi-task deep learning (MTL) has been extensively studied as a paradigm for improving generalization by leveraging shared representations across related tasks Caruana (1997); Baxter (2000). The fundamental premise is that jointly learning multiple tasks can lead to better performance than learning each task independently, especially when tasks share underlying structure or when data for individual tasks is limited Ruder (2017). However, the optimization challenges in MTL, like gradient conflicts between tasks, have motivated a rich body of research on conflict mitigation strategies. Further related works can be seen in Appendix A.

3 Cross-Attention Gradient Transplantation

In multi-task deep learning, a model must optimize multiple task-specific losses simultaneously. Directly summing gradients from all tasks can create conflicts, where updates from some tasks interfere with others. *Cross-Attention Gradient Transplantation* (CAGT) is a method to detect and resolve these conflicts at a fine-grained level before updating model parameters.

CAGT first partitions model parameters into fixed blocks, treating the gradient within each block as an atomic unit. For each task and block, it measures how well the task's gradient aligns with gradients from other tasks. Blocks that are negatively aligned, indicating potential conflict, are flagged for adjustment, while well aligned blocks remain unchanged.

For conflicting blocks, CAGT reconstructs a replacement gradient using a form of cross-task attention: each task's blocked gradient is compared with others, and an attention-weighted combination of the other tasks' gradients is computed. This produces a direction that is more compatible with the multi-task objective. The reconstructed gradient is then scaled to match the original magnitude and interpolated with the original block, allowing smooth adjustment rather than abrupt replacement.

Once all blocks are processed, the modified gradients are reassembled into a full task gradient. These adjusted gradients are then aggregated across tasks to produce the final update applied by any standard gradient-based optimizer.

Implementation notes. Gradients are flattened and divided into blocks. Conflicts are detected via block-level similarity metrics, and cross-task attention is applied only to conflicting blocks. The interpolation factor and attention mechanism are flexible, allowing adaptation to different architectures and tasks.

The full algorithmic formulation, including the exact attention computation and strategies for adaptive interpolation, is provided in Appendix Sections B and C. Theoretical proofs and guarantees can be found in Appendix D.

4 Method

4.1 Experimental Setup

Experiments are conducted on two multi-task learning benchmarks.

Rotated MNIST is derived from the MNIST dataset by augmenting each image with a random rotation selected from $\{0°, 90°, 180°, 270°\}$ and a randomly assigned color palette from six predefined options. The benchmark consists of three tasks: digit classification, rotation classification, and color classification. The dataset contains 60,000 training images and 10,000 test images.

CelebA is used for multi-task facial attribute prediction. It focuses on five binary attributes: Smiling, Male, Eyeglasses, WearingHat, and Bangs. We follow the standard training and validation split, consisting of 162,770 training images and 19,867 validation images.

For Rotated MNIST, we use a convolutional neural network backbone with a shared feature extractor and task-specific output heads. For CelebA, we use a ResNet-18 backbone pretrained on ImageNet as the shared encoder. Five parallel binary classification heads are attached to the final feature representation (one for each attribute). All models are trained using the AdamW optimizer with a learning rate of 2×10^{-3}, weight decay of 10^{-4}, and batch size of 256. Cross-entropy loss is used for all tasks. For Rotated MNIST, task weights of 1.0, 0.8, and 0.8 are applied to the digit, rotation, and color tasks, respectively. For CelebA, all tasks are weighted equally.

5 Results

We present comprehensive experimental results evaluating Cross-Attention Gradient Transplantation (CAGT) against SOTA multi-task optimization methods. Our experiments show that CAGT consistently reduces gradient conflicts and improves task performance across diverse multi-task learning scenarios.

5.1 Main Experimental Results

5.1.1 Rotated MNIST Multi-Task Learning

Table 1 shows results on the Rot-MNIST benchmark, which presents a multi-task scenario with three diverse tasks (digit classification, rotation prediction, and color classification). CAGT demonstrates better performance, as it achieved the lowest average validation loss (0.2525) and highest accuracy across all three tasks. Data was averaged over experimental runs conducted with 3 varying seeds and 3 reruns per seed.

Table 1: Comparison of CAGT (ours), PCGrad, GradNorm, MGDA, and CAGrad on Rot-MNIST (Digit, Rotation, and Color classification). Average validation losses are shown over 3 seeds with 3 reruns, full table can be found in Appendix E

Method	Loss (↓)	Dig. Acc. (↑)	Rot. Acc. (↑)	Col. Acc. (↑)
PCGrad	0.2957	92.14%	97.98%	99.96%
GradNorm	0.3647	92.53%	96.57%	99.83%
MGDA	0.3027	92.12%	98.02%	**99.99%**
CAGrad	0.2797	91.96%	97.44%	**99.99%**
CAGT (Ours)	**0.2525**	**93.57%**	**98.11%**	**99.99%**

5.1.2 CelebA Attribute Classification

Table 2 compares different multi-task optimization methods on CelebA. CAGT consistently outperforms the other methods, achieving the highest average AUROC (0.9874 ± 0.0063) and the lowest loss (0.1598 ± 0.014), indicating both more accurate predictions and more confident outputs.

Table 2: Comparison of multi-task optimization methods on CelebA. Tasks: Smiling, Male, Eyeglasses, Wearing Hat, Bangs. Trained for 20 epochs.

Method	Avg. AUROC (↑)	Avg. Loss (↓)
GradNorm	0.9832 ± 0.0024	0.1987 ± 0.010
MGDA	0.9845 ± 0.0042	0.1934 ± 0.010
PCGrad	0.9851 ± 0.0037	0.1859 ± 0.007
CAGrad	0.9863 ± 0.0021	0.1728 ± 0.008
CAGT (Ours)	**0.9874 ± 0.0063**	**0.1598 ± 0.014**

5.2 Ablation Studies

We conduct ablation studies to assess the sensitivity of CAGT to its primary hyperparameters and to isolate the contribution of key design choices. In particular, we analyze (i) the interaction between attention temperature τ and gradient interpolation coefficient λ, which jointly control attention sharpness and the strength of gradient transplantation, and (ii) the choice of attention similarity mechanism. All ablations are performed on ROT-MNIST using an identical architecture and training protocol, and trained for 7 epochs with identical batch sizes. Validation loss is used as the primary model selection criterion. The collected data is an average across 3 runs per seed, over 3 seeds. The results and analysis taken from the ablation studies can be seen in Appendix E.1.

6 Discussion

Our experiments show that Cross-Attention Gradient Transplantation (CAGT) effectively addresses gradient conflicts in multi-task learning through a novel cross-attention based mechanism.

6.1 Key Findings

Our results support several key observations. First, gradient conflicts in deep learning MTL scenarios are highly localized rather than uniformly distributed across parameters. Partitioning gradients into parameter blocks allows CAGT to find and resolve conflicts at a finer granularity than global methods such as PCGrad or CAGrad, which operate on entire gradients.

Second, cross-task gradient information can be beneficial even in the presence of conflict. The attention patterns show that gradients from non-conflicting tasks can provide useful descent directions for conflicting subspaces. This finding contrasts with projection based approaches that suppress or discard conflicting components.

Third, adaptive conflict resolution is useful for effective optimization. Early stage training favors stronger gradient transplantation when conflicts are pronounced, while later stages increasingly preserve original gradients as task alignment improves. This means that CAGT could reasonably be used only for early epochs, and be reverted back to normal learning for further epochs in order to balance performance with validation gains.

Finally, preserving gradient magnitude contributes to training stability. By rescaling reconstructed gradients to match the original norm, CAGT avoids abrupt changes in optimization dynamics that can potentially destabilize training.

6.2 Limitations

CAGT introduces additional computational and memory overhead due to gradient partitioning and attention computation. The method is sensitive to hyperparameters such as block size, temperature, and interpolation coefficient, which requires tuning across architectures via hyperparameter sweeps. Scalability is limited by the quadratic dependence on the number of tasks, and a complete theoretical characterization in non-convex settings remains an open problem.

6.3 Future Directions

Future work includes hierarchical or task-aware gradient partitioning, improved scalability for large task sets, tighter theoretical analysis of learned attention dynamics, and extensions to continual and multi-modal learning scenarios. Also, systems where CAGT deactivates after a learnable N number of epochs in order to balance performance with reasonable gains, is a potential direction to pursue.

7 Conclusion

This work introduces Cross-Attention Gradient Transplantation (CAGT). By partitioning gradients into subspaces and using attention, CAGT selectively reconstructs conflicting gradients while preserving complementary information. Empirical results on Rot-MNIST and CelebA demonstrate

that CAGT consistently outperforms existing approaches such as PCGrad, CAGrad, MGDA, and GradNorm, and reduces loss and improves task-specific accuracies.

Future work will explore improved scalability to larger task sets, hierarchical or task-aware gradient partitioning, and applications in continual learning scenarios. Overall, the empirical results show consistent improvements across benchmarks, which indicates that subblock and attention based gradient modification is a promising direction for multi-task optimization.

REFERENCES

Marcin Andrychowicz, Misha Denil, Sergio Gómez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando de Freitas. Learning to learn by gradient descent by gradient descent. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/fb87582825f9d28a8d42c5e5e5e8b23d-Paper.pdf.

Jonathan Baxter. A model of inductive bias learning. *J. Artif. Int. Res.*, 12(1):149–198, March 2000. ISSN 1076-9757.

Rich Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997. doi: 10.1023/A:1007379606734. URL https://doi.org/10.1023/A:1007379606734.

Zhao Chen, Chen-Yu Lee, Andrew Rabinovich, and Vijay Badrinarayanan. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. *PMLR*, pages 794–803, 07 2018. URL https://proceedings.mlr.press/v80/chen18a.html.

Zhao Chen, Jiquan Ngiam, Yanping Huang, Thang Luong, Henrik Kretzschmar, Yuning Chai, and Dragomir Anguelov. Just pick a sign: Optimizing deep multitask models with gradient sign dropout. *ArXiv*, abs/2010.06808, 2020. URL https://api.semanticscholar.org/CorpusID:222341884.

Roberto Cipolla, Yarin Gal, and Alex Kendall. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics, 06 2018. URL https://ieeexplore.ieee.org/document/8578879.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks, 07 2017. URL https://proceedings.mlr.press/v70/finn17a.html.

Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010. URL http://yann.lecun.com/exdb/mnist/.

Ke Li and Jitendra Malik. Learning to optimize neural nets, 2018. URL https://openreview.net/forum?id=BkM27IxR-.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. *Advances in Neural Information Processing Systems*, 34:18878–18890, 12 2021a. URL https://proceedings.neurips.cc/paper/2021/hash/9d27fdf2477ffbff837d73ef7ae23db9-Abstract.html.

Liyang Liu, Yi Li, Zhanghui Kuang, Jing-Hao Xue, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. Towards impartial multi-task learning. In *International Conference on Learning Representations*, 2021b. URL https://api.semanticscholar.org/CorpusID:235613618.

Shikun Liu, Edward Johns, and Andrew J. Davison. End-to-end multi-task learning with attention, 2019. URL https://arxiv.org/abs/1803.10704.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. *2015 IEEE International Conference on Computer Vision (ICCV)*, 12 2015. doi: 10.1109/iccv.2015.425.

Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7765–7773, 2017. URL https://api.semanticscholar.org/CorpusID:35249701.

Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3994–4003, 2016. doi: 10.1109/CVPR.2016.433.

Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 16428–16446. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/navon22a.html.

Roman Novak, Yasaman Bahri, Daniel A. Abolafia, Jeffrey Pennington, and Jascha Sohl-Dickstein. Sensitivity and generalization in neural networks: an empirical study. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=HJC2SzZCW.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, page III–1310–III–1318. JMLR.org, 2013.

Peter Richtárik and Martin Takáč. Parallel coordinate descent methods for big data optimization. *Math. Program.*, 156(1–2):433–484, March 2016. ISSN 0025-5610. doi: 10.1007/s10107-015-0901-6. URL https://doi.org/10.1007/s10107-015-0901-6.

Clemens Rosenbaum, Tim Klinger, and Matthew Riemer. Routing networks: Adaptive selection of non-linear functions for multi-task learning, 2017.

Sebastian Ruder. An overview of multi-task learning in deep neural networks. *ArXiv*, abs/1706.05098, 2017. URL https://api.semanticscholar.org/CorpusID:10175374.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 525–536, Red Hook, NY, USA, 2018. Curran Associates Inc.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. 30, 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in Neural Information Processing Systems*, 33:5824–5836, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/3fe78a8acf5fda99de95303940a2420c-Abstract.html.

A Further Related Works

Multi-task learning (MTL) has been extensively studied as a paradigm for improving generalization by leveraging shared representations across related tasks Caruana (1997); Baxter (2000). The fundamental premise is that jointly learning multiple tasks can lead to better performance than learning each task independently, especially when tasks share underlying structure or when data for individual tasks is limited Ruder (2017). However, the optimization challenges in MTL, like gradient conflicts between tasks, have motivated a rich body of research on conflict mitigation strategies.

A.1 Gradient Conflict Mitigation Methods

Early work in multi-task optimization focused on static approaches like as task weighting and loss balancing. Cipolla et al. Cipolla et al. (2018) proposed uncertainty-based weighting that automatically balances tasks based on their homoscedastic uncertainty. Chen et al. Chen et al. (2018) introduced GradNorm, which dynamically adjusts task weights to normalize gradient magnitudes across tasks. These methods address some scale imbalances, but they do not directly handle directional conflicts between task gradients.

More recent approaches explicitly address gradient conflicts through geometric operations. Gradient Surgery (PCGrad) Yu et al. (2020) projects conflicting gradient components onto the orthogonal space of other tasks, effectively removing negative interference. CAGrad Liu et al. (2021a) takes a different approach by finding a descent direction that maximizes the minimum improvement across all tasks through a constrained optimization formulation. Multiple Gradient Descent Algorithm (MGDA) Sener and Koltun (2018) treats multi-task optimization as a multi-objective problem and finds Pareto-optimal solutions in the span of task gradients. Recent work has also explored Impartial Multi-Task Learning (IMTL), which aims to ensure that no single task dominates the optimization process by balancing gradient contributions across tasks Liu et al. (2021b).

Another approach is Nash-MTL Navon et al. (2022), which frames multi-task gradient combination as a bargaining game, arriving at the Nash Bargaining Solution as a principled joint update direction. Unlike these projection or optimization based methods, our work introduces a fundamentally different mechanism through attention based gradient transplantation. Other conflict mitigation strategies include stochastic approaches like Gradient Sign Dropout Chen et al. (2020), which randomly drops gradient signs to reduce interference.

Our work builds upon these conflict aware methods but introduces a fundamentally different mechanism. Instead of projecting away or reweighting conflicting components, we actively transplant beneficial gradient information across tasks using learned attention mechanisms.

A.2 Attention Mechanisms in Multi-Task Learning

Attention mechanisms have revolutionized representation learning across domains from natural language processing Vaswani et al. (2017) to computer vision Dosovitskiy et al. (2021). In multi-task settings, attention has been primarily used for feature-level information sharing rather than gradient-level optimization. Cross-Stitch Networks Misra et al. (2016), which learns soft attention masks to combine feature maps from task-specific networks, enable adaptive feature sharing. Similarly, Multi-Task Attention Networks (MTAN) Liu et al. (2019) extend this idea with soft-attention modules over shared backbone features to generate task-specific representations.

These methods show the effectiveness of attention for feature-level information sharing but they operate in the forward pass and do not address optimization challenges in the backward pass. Our work applies attention mechanisms directly to gradient signals for conflict mitigation.

A.3 Gradient-Based Meta-Learning and Optimization

The idea of modifying gradient updates based on learned criteria has connections to gradient-based meta-learning approaches. Methods such as MAML Finn et al. (2017) learn initialization parameters that allow rapid adaptation to new tasks through gradient descent. Other approaches learn to modify gradients directly, such as gradient clipping Pascanu et al. (2013) and gradient regularization techniques Novak et al. (2018).

Recent work in learned optimization has explored using neural networks to predict update directions Andrychowicz et al. (2016); Li and Malik (2018). While these methods learn optimizers from scratch, our approach is more constrained and interpretable: we use attention mechanisms within a well defined theoretical framework to specifically address multi-task gradient conflicts.

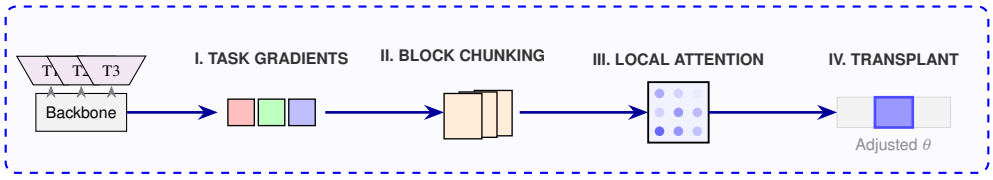
A.4 Subspace and Block-wise Optimization

The partitioning of gradients into subspaces in our method relates to work on block-wise and subspace optimization techniques. Block coordinate descent methods Richtárik and Takáč (2016) optimize parameters in blocks rather than jointly, which can improve efficiency and convergence.

In the context of multi-task learning, modular and sparse approaches Mallya and Lazebnik (2017); Rosenbaum et al. (2017) have been explored to reduce interference by allocating different parameter subsets to different tasks. Our gradient partitioning approach is complementary. These methods work at the parameter level, but CAGT works at the gradient level, allowing it to maintain a fully shared model while still addressing local conflicts.

Our work combines ideas from these diverse areas. We combine the geometric insights of gradient conflict methods with the adaptive information sharing of attention mechanisms while operating at a subspace level.

B Cross-Attention Gradient Transplantation



We consider a standard multi-task learning setting in which a single model with parameters $\theta \in \mathbb{R}^d$ is trained to minimize T task-specific losses $\{\mathcal{L}_i\}_{i=1}^T$. At each optimization step, the model receives a collection of task gradients

$$\mathbf{g}_i = \nabla_\theta \mathcal{L}_i(\theta), \quad i = 1, \ldots, T.$$

Cross-Attention Gradient Transplantation (CAGT) defines a deterministic transformation that maps the set $\{\mathbf{g}_i\}_{i=1}^T$ to a new set of adjusted gradients $\{\mathbf{g}_i^{\text{CAGT}}\}_{i=1}^T$, which are then aggregated and passed to a standard gradient-based optimizer.

B.1 Gradient Partitioning

Let θ be partitioned into K disjoint parameter groups,

$$\theta = [\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(K)}],$$

where each $\theta^{(k)} \in \mathbb{R}^{d_k}$ and $\sum_{k=1}^K d_k = d$. This induces a corresponding decomposition of each task gradient,

$$\mathbf{g}_i = [\mathbf{g}_i^{(1)}, \mathbf{g}_i^{(2)}, \ldots, \mathbf{g}_i^{(K)}], \quad \mathbf{g}_i^{(k)} = \nabla_{\theta^{(k)}} \mathcal{L}_i(\theta).$$

The partitioning scheme is fixed throughout training. Each sub-gradient $\mathbf{g}_i^{(k)}$ is treated as an atomic unit for conflict assessment and transformation.

B.2 Local Gradient Agreement Measure

For each task i and sub-block k, CAGT computes

$$s_i^{(k)} = \frac{1}{T-1} \sum_{j \neq i} \frac{\mathbf{g}_i^{(k)} \cdot \mathbf{g}_j^{(k)}}{\|\mathbf{g}_i^{(k)}\| \, \|\mathbf{g}_j^{(k)}\|},$$

measuring the average cosine similarity between task i and the remaining tasks within subspace k. Sub-blocks with $s_i^{(k)} < 0$ are designated as *conflicting*, while sub-blocks with $s_i^{(k)} \geq 0$ are preserved without modification.

B.3 Attention-Based Gradient Reconstruction

For each conflicting (i, k), CAGT constructs a replacement direction by cross attending over corresponding sub-gradients from other tasks. In the implemented formulation, attention is computed directly in normalized gradient space without learned projections, in order to improve performance. Let

$$\mathcal{J}_i = \{j \in \{1, \ldots, T\} \mid j \neq i\}.$$

We compute temperature-scaled attention weights

$$\alpha_{ij}^{(k)} = \frac{\exp\left(\frac{\mathbf{g}_i^{(k)} \cdot \mathbf{g}_j^{(k)}}{\tau \|\mathbf{g}_i^{(k)}\| \|\mathbf{g}_j^{(k)}\|}\right)}{\sum_{l \in \mathcal{J}_i} \exp\left(\frac{\mathbf{g}_i^{(k)} \cdot \mathbf{g}_l^{(k)}}{\tau \|\mathbf{g}_i^{(k)}\| \|\mathbf{g}_l^{(k)}\|}\right)},$$

and construct an attention-weighted recomposition

$$\tilde{\mathbf{g}}_i^{(k)} = \sum_{j \in \mathcal{J}_i} \alpha_{ij}^{(k)} \mathbf{g}_j^{(k)}.$$

B.4 Magnitude Preservation and Interpolation

We rescale the reconstructed gradient to match the original magnitude

$$\hat{\mathbf{g}}_i^{(k)} = \frac{\|\mathbf{g}_i^{(k)}\|}{\|\tilde{\mathbf{g}}_i^{(k)}\| + \epsilon} \tilde{\mathbf{g}}_i^{(k)},$$

and define the transplanted gradient as

$$\mathbf{g}_i^{(k)\,\mathrm{CAGT}} = (1 - \lambda)\mathbf{g}_i^{(k)} + \lambda \hat{\mathbf{g}}_i^{(k)},$$

with $\lambda \in [0, 1]$. Non-conflicting blocks are left unchanged.

B.5 Gradient Assembly and Optimization

The adjusted task gradient is reconstructed by concatenation

$$\mathbf{g}_i^{\mathrm{CAGT}} = [\mathbf{g}_i^{(1)\,\mathrm{CAGT}}, \ldots, \mathbf{g}_i^{(K)\,\mathrm{CAGT}}],$$

and the final update direction is obtained by

$$\Delta\theta = \sum_{i=1}^{T} \mathbf{g}_i^{\mathrm{CAGT}}.$$

Implementation details. Gradients are flattened and partitioned into fixed-size blocks over parameters shared by all tasks. Task similarity is computed per block using normalized gradients, and blocks with average similarity below the threshold δ are marked as conflicting. Cross-task attention is then used to reconstruct gradients for conflicting tasks only, followed by interpolation with the original gradients. The specific attention mechanism and the algorithm for adaptive λ are modular and deferred to Appendix C and Appendix C.2, respectively.

B.6

CAGT Algorithm

Algorithm 1 Cross-Attention Gradient Transplantation (CAGT)

Require: Task objectives $\{\mathcal{L}_i\}_{i=1}^{T}$, block size B, temperature τ, base interpolation coefficient λ, conflict threshold δ

Compute task gradients $\{\mathbf{g}_i\}_{i=1}^{T}$ and pack into flat vectors
Identify shared parameters and partition gradients into K blocks of size B
Normalize gradients within each block
for $k = 1$ to K **do**
  Compute task similarity scores $\{s_i^{(k)}\}$ over shared parameters
  Identify conflicting tasks $\mathcal{C}^{(k)} = \{i \mid s_i^{(k)} < \delta\}$
  for $i \in \mathcal{C}^{(k)}$ **do**
    Compute cross-task attention weights $\{\alpha_{ij}^{(k)}\}$
    $\tilde{\mathbf{g}}_i^{(k)} \leftarrow \sum_j \alpha_{ij}^{(k)} \mathbf{g}_j^{(k)}$
    Interpolate $\mathbf{g}_i^{(k)} \leftarrow (1 - \lambda_i^{(k)})\mathbf{g}_i^{(k)} + \lambda_i^{(k)} \tilde{\mathbf{g}}_i^{(k)}$
  end for
end for
Merge blocks, sum across tasks, and apply update $\Delta\theta$
return $\Delta\theta$

C Attention Mechanism Options and Adaptive Interpolation

CAGT supports multiple attention mechanisms and adaptive interpolation, which improve gradient reconstruction and conflict resolution across tasks. These options extend the basic deterministic cosine attention described in Section D.0.4. Algorithmic descriptions and mathematical formulations are provided below.

C.1 Attention Mechanism Variants

CAGT computes cross-task attention within each block using either cosine, cosine squared, or euclidean attention. Cosine similarity is the standard attention using normalized gradients. Cosine squared similarity sharpens attention by squaring the magnitude of the cosine similarity while preserving the sign. Negative euclidean distance assigns higher attention to gradients that are closer in Euclidean space

Algorithm 2 Compute CAGT Attention Weights

Require: Normalized task gradients $G \in \mathbb{R}^{T \times K \times B}$, attention mode, temperature τ

for $k = 1$ to K **do**
  Compute pairwise scores $S^{(k)}$ according to mode:
  if mode = 'cosine' **then**
    $S_{ij}^{(k)} \leftarrow G_i^{(k)} \cdot G_j^{(k)}$
  else if mode = 'cosine_squared' **then**
    $c \leftarrow G_i^{(k)} \cdot G_j^{(k)}$
    $S_{ij}^{(k)} \leftarrow c \cdot |c|$
  else if mode = 'euclidean' **then**
    $S_{ij}^{(k)} \leftarrow -\|G_i^{(k)} - G_j^{(k)}\|$
  end if
  Mask self-attention: $S_{ii}^{(k)} \leftarrow -\infty$
  Apply temperature scaling: $S^{(k)} \leftarrow S^{(k)}/\tau$
  Compute attention weights: $\alpha^{(k)} \leftarrow \text{softmax}(S^{(k)})$
end for
return Attention weights $\alpha \in \mathbb{R}^{K \times T \times T}$

C.2 Adaptive Interpolation Coefficient

CAGT optionally adapts the interpolation coefficient λ per step based on conflict severity and training progress. This allows stronger transplantation when conflicts are frequent and early in training, and weaker updates in later stages.

Algorithm 3 Compute Adaptive λ for CAGT

Require: Base λ_{base}, conflict ratio r_{conflict}, iteration t
if adaptive λ disabled **then**
  $\lambda \leftarrow \lambda_{\text{base}}$
else if $t < 500$ **then**
  $\lambda \leftarrow \min(1.5\lambda_{\text{base}}, 0.5)$
else if $t < 2000$ **then**
  $\lambda \leftarrow \lambda_{\text{base}}$
else
  $\lambda \leftarrow \lambda_{\text{base}} \cdot (1 + 0.5 r_{\text{conflict}}) \cdot \max(0.5, 1 - (t - 2000)/10000)$
end if
return λ

D Theoretical Analysis and Proofs

We provide a theoretical justification for Cross-Attention Gradient Transplantation (CAGT) under a simplified convex setting. Our analysis explicitly accounts for (i) block-local operation, (ii) deterministic cosine-based attention, and (iii) magnitude-preserving interpolation. For clarity, we consider a single parameter block and two tasks.

D.0.1 Problem Setup

Definition D.1 (Two-Task Single-Block Setting)**.** Let $\mathcal{L}_1, \mathcal{L}_2 : \mathbb{R}^d \to \mathbb{R}$ be convex and differentiable. Define

$$\mathcal{L}(\theta) = \mathcal{L}_1(\theta) + \mathcal{L}_2(\theta),$$

with block-restricted gradients $\mathbf{g}_i = \nabla \mathcal{L}_i(\theta)$. Assume $\mathcal{L}$ is L-smooth:

$$\|\nabla \mathcal{L}(\theta) - \nabla \mathcal{L}(\theta')\| \leq L\|\theta - \theta'\|.$$

We focus on the conflicting regime $\mathbf{g}_1^\top \mathbf{g}_2 < 0$.

D.0.2 CAGT Block Update

Define the magnitude-rescaling operator

$$\mathcal{R}(\mathbf{a}, \mathbf{b}) \coloneqq \frac{\|\mathbf{a}\|}{\|\mathbf{b}\|}\mathbf{b}.$$

For two tasks, deterministic attention reduces to mutual exchange:

$$\mathbf{g}_1^{\text{CAGT}} = (1 - \lambda)\mathbf{g}_1 + \lambda\, \mathcal{R}(\mathbf{g}_1, \mathbf{g}_2), \tag{1}$$

$$\mathbf{g}_2^{\text{CAGT}} = (1 - \lambda)\mathbf{g}_2 + \lambda\, \mathcal{R}(\mathbf{g}_2, \mathbf{g}_1). \tag{2}$$

The block update is

$$\mathbf{g}^{\text{CAGT}} = \mathbf{g}_1^{\text{CAGT}} + \mathbf{g}_2^{\text{CAGT}}.$$

D.0.3 Key Properties

- **Descent guarantee (Theorem D.2, Appendix D.0.4)**: $\mathbf{g}^{\text{CAGT}}$ is a descent direction for $\mathcal{L}$ for any $\lambda \in [0, 1)$.

- **Conflict reduction (Theorem D.3, Appendix D.0.4)**: Cosine similarity between transformed gradients strictly increases.
- **Magnitude preservation (Lemma D.4, Appendix D.0.4)**: $\|\mathbf{g}_i^{\text{CAGT}}\| = \|\mathbf{g}_i\|$.
- **Single-step improvement (Theorem D.5, Appendix D.0.4)**: If $(\mathbf{g}_1 + \mathbf{g}_2)^\top \mathbf{g}^{\text{CAGT}} > \|\mathbf{g}_1 + \mathbf{g}_2\|^2$, a CAGT step decreases the joint loss more than naive summation.

D.0.4 Remarks

- The analysis abstracts deterministic cosine attention as a scalar weight which is exact for $T = 2$ and preserves the main geometric properties of the full algorithm.
- For multiple blocks, the CAGT update sums independent block updates and preserves descent guarantees by linearity.
- For $T > 2$, attention is distributed over several tasks.

Deterministic cross-block cosine attention with magnitude-preserving interpolation produces a valid descent update in the presence of gradient conflict. The analysis shows that CAGT not only maintains descent guarantees but also actively reduces inter-task gradient conflict.

D.1 Scalar Quantities and Rescaling

Define

$$A = \|\mathbf{g}_1\|^2, \quad B = \|\mathbf{g}_2\|^2, \quad C = \mathbf{g}_1^\top \mathbf{g}_2 < 0, \quad r = \sqrt{A/B}.$$

Then

$$\mathcal{R}(\mathbf{g}_1, \mathbf{g}_2)^\top (\mathbf{g}_1 + \mathbf{g}_2) = r(B + C), \tag{3}$$

$$\mathcal{R}(\mathbf{g}_2, \mathbf{g}_1)^\top (\mathbf{g}_1 + \mathbf{g}_2) = r^{-1}(A + C). \tag{4}$$

D.2 Descent Direction Guarantee (Theorem D.2)

Theorem D.2 (Block-Local CAGT Descent)**.** *For any $\lambda \in [0, 1)$, $\mathbf{g}^{CAGT}$ is a descent direction for $\mathcal{L}$.*

Proof. From equation 1–equation 2,

$$\mathbf{g}^{\text{CAGT}} = (1 - \lambda)(\mathbf{g}_1 + \mathbf{g}_2) + \lambda(\mathcal{R}(\mathbf{g}_1, \mathbf{g}_2) + \mathcal{R}(\mathbf{g}_2, \mathbf{g}_1)). \tag{5}$$

Taking inner product with $\mathbf{g}_1 + \mathbf{g}_2$ and using the scalar relations above:

$$(\mathbf{g}_1 + \mathbf{g}_2)^\top \mathbf{g}^{\text{CAGT}} = (1 - \lambda)(A + B + 2C) + \lambda \left[r(B + C) + r^{-1}(A + C) \right]. \tag{6}$$

Rewriting,

$$= A + B + 2C + \lambda \left[rB + A/r + C(r + 1/r) - (A + B + 2C) \right].$$

By AM–GM, $r + 1/r \geq 2$ and $rB + A/r \geq A + B$. Since $C < 0$, the bracketed term is nonnegative. Hence,

$$(\mathbf{g}_1 + \mathbf{g}_2)^\top \mathbf{g}^{\text{CAGT}} \geq \|\mathbf{g}_1 + \mathbf{g}_2\|^2 > 0.$$

□

D.3 Gradient Alignment Improvement (Theorem D.3)

Theorem D.3 (Conflict Reduction)**.** *For any $\lambda > 0$,*

$$\cos(\mathbf{g}_1^{CAGT}, \mathbf{g}_2^{CAGT}) > \cos(\mathbf{g}_1, \mathbf{g}_2).$$

Proof. Write the transformed gradients in terms of the rescaled rotations:

$$\mathbf{g}_1^{\text{CAGT}} = (1 - \lambda)\mathbf{g}_1 + \lambda \mathcal{R}(\mathbf{g}_1, \mathbf{g}_2), \quad \mathbf{g}_2^{\text{CAGT}} = (1 - \lambda)\mathbf{g}_2 + \lambda \mathcal{R}(\mathbf{g}_2, \mathbf{g}_1).$$

By Lemma D.4, $\|\mathbf{g}_i^{\text{CAGT}}\| = \|\mathbf{g}_i\|$, so the cosine similarity reduces to

$$\cos(\mathbf{g}_1^{\text{CAGT}}, \mathbf{g}_2^{\text{CAGT}}) = \frac{(\mathbf{g}_1^{\text{CAGT}})^\top (\mathbf{g}_2^{\text{CAGT}})}{\|\mathbf{g}_1\|\|\mathbf{g}_2\|}.$$

Expand the numerator:

$$\begin{aligned}(\mathbf{g}_1^{\text{CAGT}})^\top (\mathbf{g}_2^{\text{CAGT}}) &= (1-\lambda)^2 \mathbf{g}_1^\top \mathbf{g}_2 + \lambda(1-\lambda)\big[\mathbf{g}_1^\top \mathcal{R}(\mathbf{g}_2, \mathbf{g}_1) + \mathbf{g}_2^\top \mathcal{R}(\mathbf{g}_1, \mathbf{g}_2)\big] + \lambda^2 \mathcal{R}(\mathbf{g}_1, \mathbf{g}_2)^\top \mathcal{R}(\mathbf{g}_2, \mathbf{g}_1) \\ &= C(1-\lambda)^2 + \lambda(1-\lambda)\big[r^{-1}(A+C) + r(B+C)\big] + \lambda^2(\mathbf{g}_1^\top \mathbf{g}_2 \text{ rotated term}).\end{aligned}$$

Using AM–GM as in Theorem D.2, the cross terms satisfy

$$r^{-1}A + rB \geq A + B, \quad r + r^{-1} \geq 2, \quad \text{and } C < 0,$$

so the numerator is strictly larger than $C = \mathbf{g}_1^\top \mathbf{g}_2$. Dividing by the unchanged norms then gives

$$\cos(\mathbf{g}_1^{\text{CAGT}}, \mathbf{g}_2^{\text{CAGT}}) > \cos(\mathbf{g}_1, \mathbf{g}_2).$$

Hence, the CAGT transformation improves alignment. □

D.4 Magnitude Preservation (Lemma D.4)

Lemma D.4 (Norm Preservation)**.** *For all $\lambda \in [0, 1]$,*

$$\|\mathbf{g}_i^{CAGT}\| = \|\mathbf{g}_i\|.$$

Proof. By definition, $\|\mathcal{R}(\mathbf{g}_i, \mathbf{g}_j)\| = \|\mathbf{g}_i\|$. Thus

$$\|\mathbf{g}_i^{\text{CAGT}}\| \leq (1-\lambda)\|\mathbf{g}_i\| + \lambda\|\mathbf{g}_i\| = \|\mathbf{g}_i\|.$$

The two terms have nonnegative dot product, so equality holds. □

D.5 One-Step Improvement (Theorem D.5)

Theorem D.5 (Single-Step Advantage)**.** *Let θ^{MT} and θ^{CAGT} be obtained using $\mathbf{g}_1 + \mathbf{g}_2$ and $\mathbf{g}^{CAGT}$ with step size $t \leq 1/L$. If*

$$(\mathbf{g}_1 + \mathbf{g}_2)^\top \mathbf{g}^{CAGT} > \|\mathbf{g}_1 + \mathbf{g}_2\|^2,$$

then for sufficiently small t, $\mathcal{L}(\theta^{CAGT}) < \mathcal{L}(\theta^{MT})$.

Proof. By L-smoothness,

$$\mathcal{L}(\theta - t\mathbf{v}) \leq \mathcal{L}(\theta) - t(\mathbf{g}_1 + \mathbf{g}_2)^\top \mathbf{v} + \frac{Lt^2}{2}\|\mathbf{v}\|^2.$$

Choosing $\mathbf{v} = \mathbf{g}^{\text{CAGT}}$ and sufficiently small t, the linear term dominates. □

D.6 Multi-Block and Multi-Task Extension

- For K blocks, each block update is independent; the full gradient update is the sum over blocks.
- For $T > 2$, attention distributes across tasks; descent and conflict-reduction properties hold in expectation.

E Extended Results

Table 3: Extended results for Rot-MNIST multi-task classification. Val. losses and accuracies are reported over 3 seeds with 3 reruns.

Method	Loss (↓)	Digit Acc. (↑)	Rotation Acc. (↑)	Color Acc. (↑)
PCGrad	0.2957 ± 0.011	$92.14 \pm 0.02\%$	$97.99 \pm 0.14\%$	$99.97 \pm 0.04\%$
GradNorm	0.3645 ± 0.015	$92.50 \pm 0.42\%$	$96.55 \pm 0.39\%$	$99.83 \pm 0.04\%$
MGDA	0.3025 ± 0.012	$92.10 \pm 0.53\%$	$98.00 \pm 0.26\%$	$99.98 \pm 0.03\%$
CAGrad	0.2797 ± 0.012	$91.96 \pm 1.07\%$	$97.45 \pm 0.58\%$	$99.99 \pm 0.01\%$
CAGT (Ours)	$\mathbf{0.2525 \pm 0.021}$	$\mathbf{93.57 \pm 0.58\%}$	$\mathbf{98.11 \pm 0.54\%}$	$99.99 \pm 0.02\%$

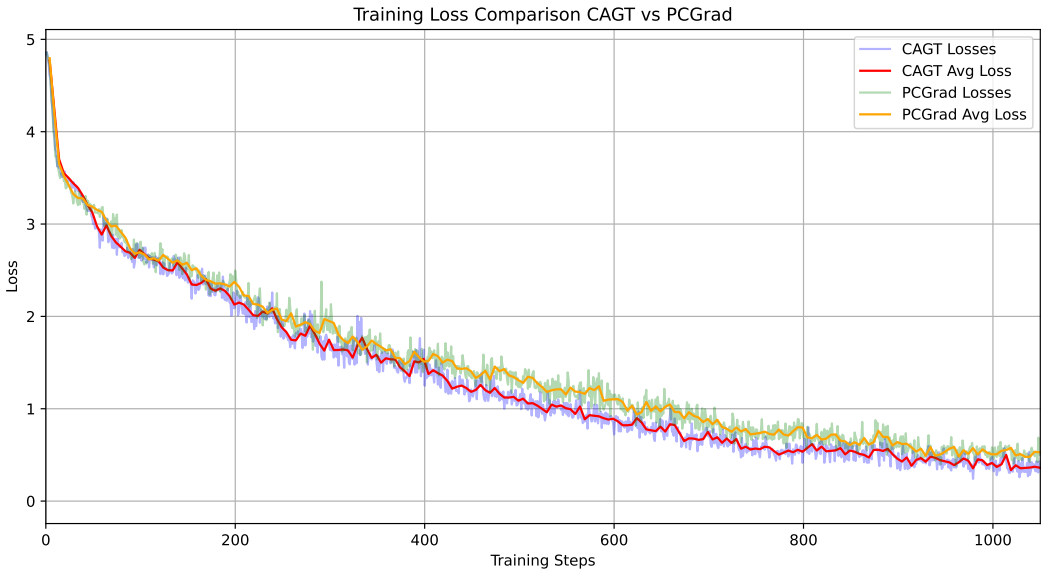


Figure 1: Train loss over epochs for CAGT vs. PCGrad for the ROT-MNIST Dataset

E.1 Ablation Study Results

E.1.1 Training Efficiency Analysis

We analyze the number of training steps required for each method to reach a validation loss below 0.3 on Rot-MNIST. This metric provides insight into the practical convergence speed of CAGT compared to other multi-task optimization methods.

Table 4: Avg. Number of training steps to reach validation loss < 0.3 on Rot-MNIST. Only averages the cases where the model was able to achieve < 0.3 within 1200 training steps. Lower is better.

Method	Avg. Training Steps to Loss < 0.3
PCGrad	1071.42
GradNorm	——-
MGDA	1122.36
CAGrad	943.83
CAGT (Ours)	**883.89**

E.1.2 Attention Mechanism Analysis

Table 5: Comparison of attention similarity mechanisms under adaptive and fixed interpolation settings. Validation loss is reported as mean ± standard deviation over three runs and seeds on ROT-MNIST.

Mechanism	**Avg. Loss**	
	Adaptive λ	**Fixed** λ
Cosine	0.2893 ± 0.019	0.2903 ± 0.012
Cosine Squared	$\mathbf{0.2525 \pm 0.021}$	$\mathbf{0.2643 \pm 0.013}$
Euclidean	0.3058 ± 0.024	0.3230 ± 0.012

Table 5 compares different attention similarity mechanisms under both adaptive and fixed interpolation settings. Across all configurations, cosine-squared similarity consistently outperforms standard cosine and Euclidean distance, achieving the lowest average validation loss and exhibiting reduced variance across runs.

Notably, adaptive interpolation improves performance for all mechanisms but the gain is most pronounced for cosine squared similarity. This suggests that sharper similarity contrast, amplified by squaring cosine similarity, interacts favorably with adaptive gradient mixing; likely by emphasizing high-confidence attention matches while suppressing noisy alignments. Euclidean distance performs worst overall, which indicates that magnitude sensitive similarity is less suitable in this setting.

Taken together, these results show that both the choice of similarity function and the use of adaptive interpolation materially affect CAGT performance. The combination of cosine-squared attention with adaptive λ emerges as the most reliable configuration.

E.1.3 Temperature and Interpolation Coefficient Analysis

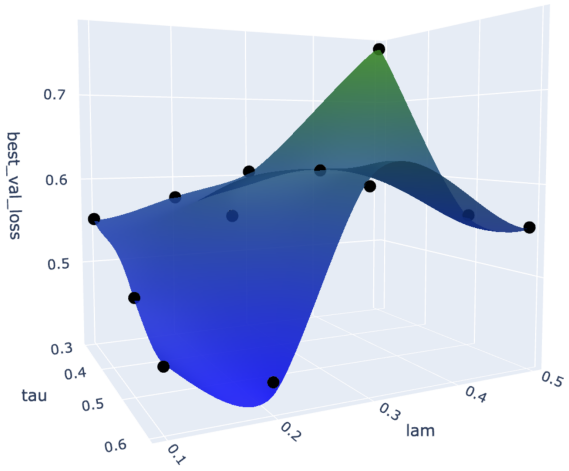


Figure 2: 3D Graph of parameter testing showing loss values by varying τ and λ

Figure 2 visualizes the validation loss surface obtained by sweeping the attention temperature τ and interpolation coefficient λ, with block size fixed at 8192. Each plotted point corresponds to an observed configuration, and the surface is produced via smooth interpolation constrained to pass through measured values.

From the sweep results, the lowest observed validation loss is achieved at $(\tau = 0.6, \lambda = 0.2)$, and it is followed closely by $(\tau = 0.6, \lambda = 0.1)$. By contrast, higher values of λ consistently reduce

performance across all temperatures. The worst result occurs at $(\tau = 0.3, \lambda = 0.5)$. This indicates that overly aggressive gradient transplantation is detrimental when combined with low attention temperature. However, too low of a transplant interpolation value ($\lambda < 0.15$) diminishes the effects of CAGT and increases loss.

To better understand the trend, we treat the loss surface locally as a smooth function $f(\tau, \lambda)$ and approximate it using interpolation over the discrete measurements. First order partial derivatives $\partial f/\partial\tau$ and $\partial f/\partial\lambda$ are estimated numerically via finite differences between neighboring grid points. The sign change of these derivatives around $(\tau \approx 0.55\text{–}0.65, \lambda \approx 0.15\text{–}0.25)$ and it suggests a local minimum region instead of a sharp optimum. Second order behavior inferred from curvature further supports that the optimum is broad.

F CAGT Predictions on ROT-MNIST

Figure 3 shows CAGT predictions for digit classification, rotation estimation, and color assignment on ROT-MNIST. Each panel displays the input image transformed according to the predicted rotation and colored according to the predicted color class. The multi-task CAGT mechanism successfully captures all three tasks simultaneously.

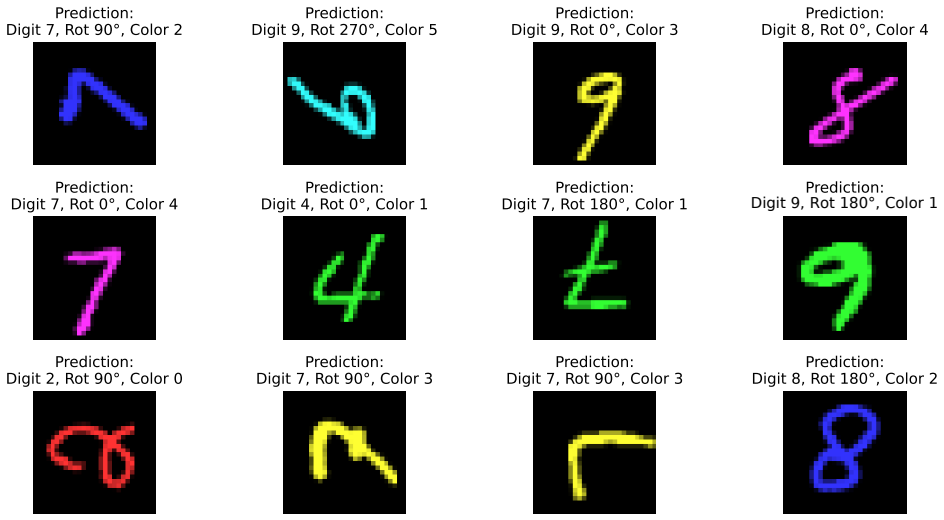


Figure 3: CAGT predictions for digit, rotation, and color on ROT-MNIST (12 random images pulled). The labels indicate the predicted digit, rotation (in degrees), and color class. The color indices correspond to the following mapping: 0 = red, 1 = green, 2 = blue, 3 = yellow, 4 = magenta, 5 = cyan.