RGB-D Domain Adaptation with Cross-Modality Self-Supervision

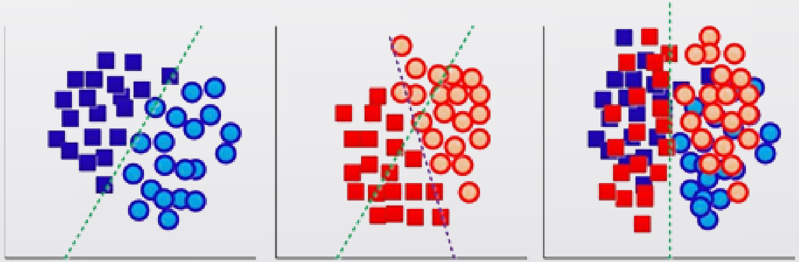
Machine learning and Deep learning
2019/2020

Hadi Nejabat

Contents

1. Addressed problem

RGB-D Domain Adaptation with
Cross-Modality Self-Supervision



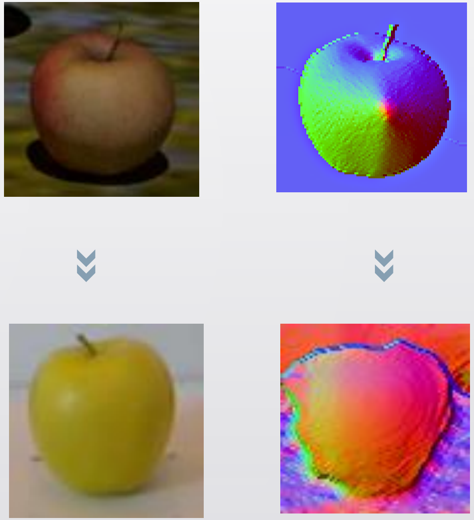
Significance:

- Domain adaptation
- Multi-modality

RGB Depth

Synthetic

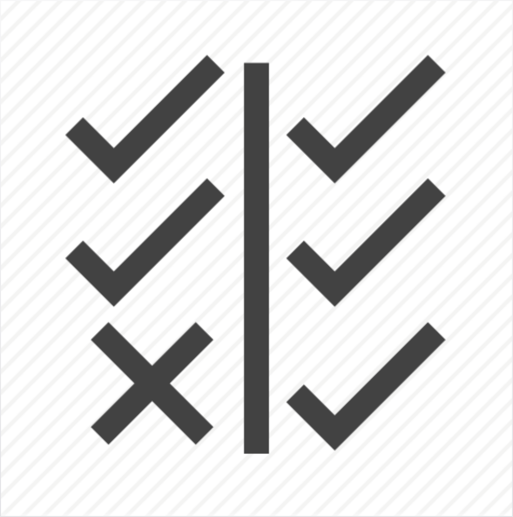
Real



2. Proposed solution

Goal:

Explore proposed solutions and compare results of variations



Datasets:

synROD → Source dataset (Synthetic origin)

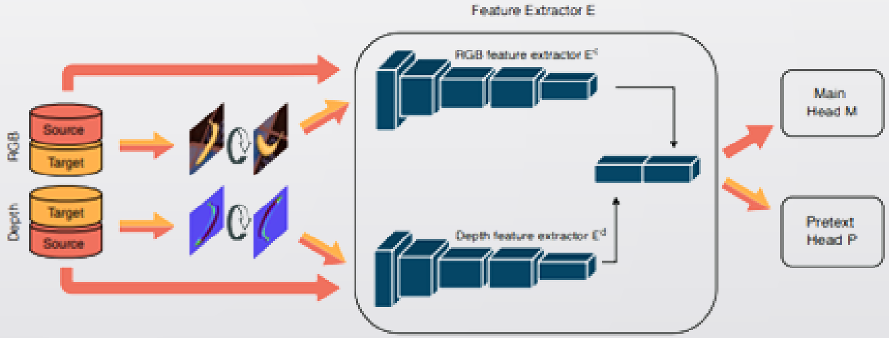
ROD → Target dataset (Real origin)

Images of each dataset are represented as RGB-D pairs.

- RGB
- Depth

Model:

- Feature extractor
- Main task
- Pretext task



End-to-end experiments:

Source only (SO)

No DA performed, used as reference baseline score.

Domain adaptation (DA)

Re-implementation

DA implemented as relative rotation between RGB-D images.

Variation

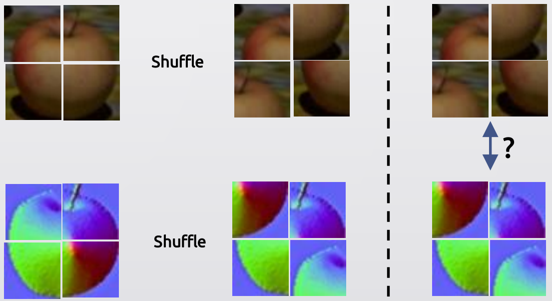
DA implemented as Jigsaw puzzle assembly between RGB-D images.

Auxiliary tasks:

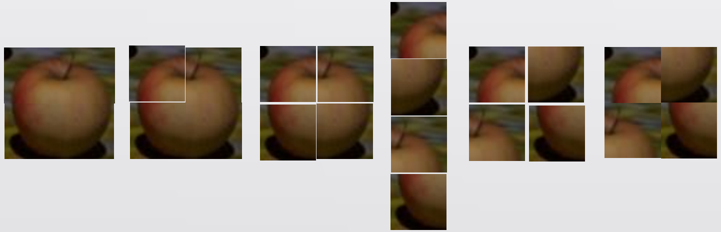
Relative rotation

Jigsaw Puzzle

(relative task)



Jigsaw puzzle details:



Algorithm 2 Jigsaw Puzzle Transformer

Input:

Image data Assigned label from DataGenerator

Output:

Transformed Image Tensor

procedure TRANSFORMING

Resize Image to 224x224

Get grid size

Generate possible permutations grid tiles

Get Minimum allowed permutation subset

Generate tiles height and width

Assign initial position to tiles

for each Image **do**

Start from initial positions

Set x,y for each tile

Create crop elements based on grid size

Extract cropped element and store

Move to next tile by updating x,y

Shuffle extracted tiles list based on permutations label

Re-assemble Images tiles

Training details:

$$Loss = Lm + \lambda p * (\hat{L}ps + \hat{L}pt)$$

	SO	DA	VAR (relative task)
Source to Main task	Cross Entropy loss	Cross Entropy loss	Cross Entropy loss
Target to Main task	N/A	Cross Entropy loss	Cross Entropy loss
Source & Target transformed to Pretext task	N/A	Cross Entropy loss	Cross Entropy loss

Algorithm 1 RGB-D Domain Adaptation

Input:
Labeled source data set $S = \{((x_i^{sc}, x_i^{sd}), y_i^s)\}_{i=1}^{Ns}$
Unlabeled target data set $T = \{((x_i^{tc}, x_i^{td})\}_{i=1}^{Nt}$
Output:
Object class prediction for the target data $\{\tilde{y}_i^t\}_{i=1}^{Nt}$

```
procedure TRAINING(S,T)
    Get transformed set S̃ = {((x̃_i^sc, x̃_i^sd), z_i^s)}_{i=1}^{Ñs}
    Get transformed set T̃ = {((x̃_i^tc, x̃_i^td), z_i^s)}_{i=1}^{Ñt}
    for each epoch do
        Load a batch from S
        Compute loss L_m
        Load a batch from S̃ and T̃
        Compute loss L_p
        Update loss L_p = *0.1 ← Regularization DA
        Update M weights with ∇L_m
        Update P weights with ∇L_p
        Update E weights with ∇L_m and ∇L̂_t
procedure TEST(T)
    for each x_i^tc, x_i^td in T do
        Compute ỹ_i^t = M(E(x_i^tc, x_i^td))
```

M = Main task, P = Pretext task, E = Feature Extractor

Other Remarks:

Optimizer: Stochastic Gradient Descent (SGD) with Momentum.

Reguralizer: Weight decay = 0.05

Drop-out: assigned with 0.5

Early-stop: assigned with patience threshold

(*) An early stopping criterion is defined
(**) Only used in DA and VAR experiments

3. Experiments and Results

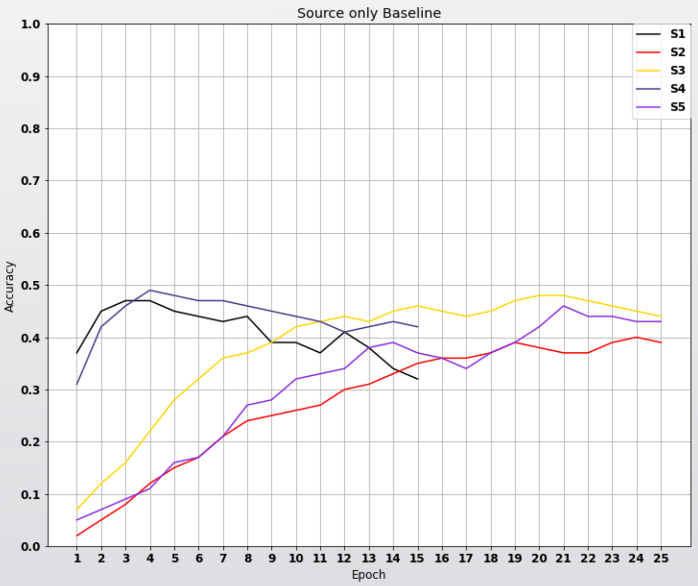
Hyperparamers:

Experiment	Param. Set	Batch size	LR	Weight loss Pretext task	Weight loss Entropy	weight_decay	dropout	Max. Validation Accuracy (%)	Diagnostics
Source only Baseline	S1	64	0.01	N/A	N/A	N/A	N/A	**48.89 @Epoch 4**	Overfitting
	S4	64	0.001	N/A	N/A	N/A	N/A	46.77 @Epoch 4	Overfitting
	S2	32	0.0001	N/A	N/A	0.005	N/A	41.2 @Epoch 18	Distrubtion due to unrepresentative classes
	S5	64	0.00001	N/A	N/A	0.05	N/A	44.36 @Epoch 19	Reuqire smoother learning
	S3	32	0.0001	N/A	N/A	0.05	0.5	47.78 @Epoch 20	Acceptable for Baseline
DA - Rotation	S7	32	0.0001	1	0.1	0.05	0.5	**61.12 @Epoch 6**	Original papers inputs
	S10	32	0.00001	1	0.1	0.05	0.5	60.06 @Epoch 9	Lower LR
	S6	32	0.0001	0.8	0.1	0.05	0.5	59.61 @Epoch 7	Lower wieght to tune objective func.
	S9	32	0.0001	1	0.1	0	0.5	59.99 @Epoch 6	No reguraizer for main task
	S8	32	0.0001	1	0.2	0.05	0.5	58.63 @Epoch 5	Not acceptable generalization
DA - Jigsaw Puzzle	S11	32	0.0001	1	0.1	0.05	0.5	**62.92 @Epoch 3**	Inputs from previous best result
	S12	32	0.0001	1	0.1	0	0.5	60.52 @Epoch 4	No reguraizer for main task
	S13	32	0.0001	0.8	0.1	0.05	0.5	61.78 @Epoch 3	Lower wieght to tune objective func.
	S14	32	0.0001	0.9	0.2	0.05	0.5	62.8 @Epoch 3	Lower wieght to tune objective func.
	S15	32	0.0001	1.5	0.1	0.05	0.5	60.4 @Epoch 3	higher wieght to tune objective func.

Hyperparameter tuning:

Source only Baseline:

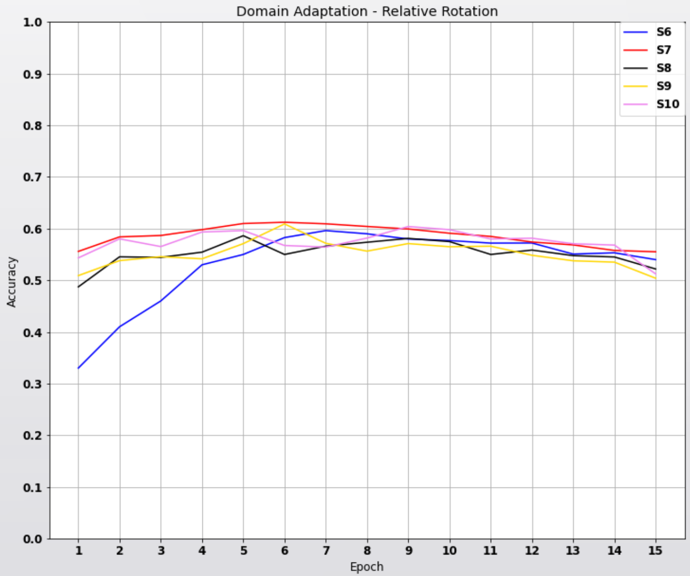
Param. Set	Batch size	LR	Weight loss Pretext task	Weight loss Entropy	weight_decay	dropout	Max. Validation Accuracy (%)
S1	64	0.01	N/A	N/A	N/A	N/A	**48.89 @Epoch 4**
S4	64	0.001	N/A	N/A	N/A	N/A	46.77 @Epoch 4
S2	32	0.0001	N/A	N/A	0.005	N/A	41.2 @Epoch 18
S5	64	0.00001	N/A	N/A	0.05	N/A	44.36 @Epoch 19
S3	32	0.0001	N/A	N/A	0.05	0.5	47.78 @Epoch 20



Hyperparameter tuning:

DA – Relative Rotation:

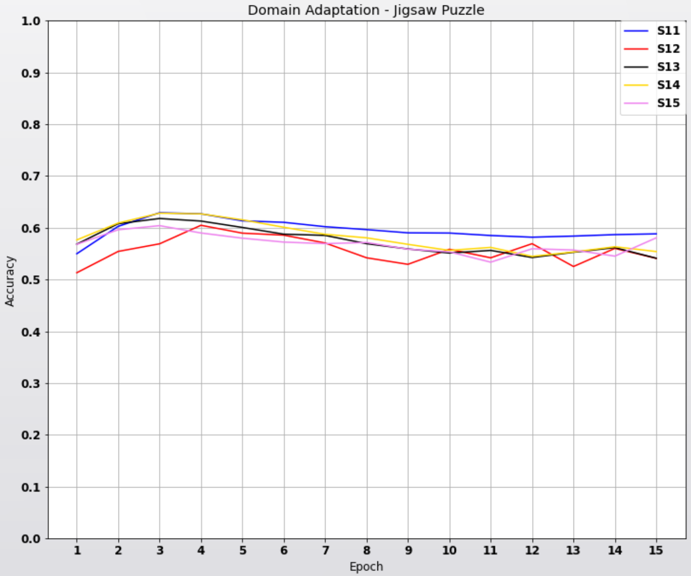
Param. Set	Batch size	LR	Weight loss Pretext task	Weight loss Entropy	weight_decay	dropout	Max. Validation Accuracy (%)
S7	32	0.0001	1	0.1	0.05	0.5	**61.12 @Epoch 6**
S10	32	0.00001	1	0.1	0.05	0.5	60.06 @Epoch 9
S6	32	0.0001	0.8	0.1	0.05	0.5	59.61 @Epoch 7
S9	32	0.0001	1	0.1	0	0.5	59.99 @Epoch 6
S8	32	0.0001	1	0.2	0.05	0.5	58.63 @Epoch 5



Hyperparameter tuning:

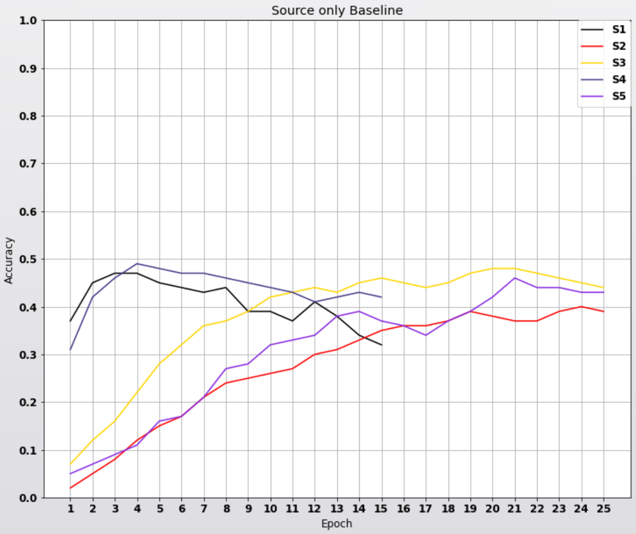
DA – Jigsaw Puzzle:

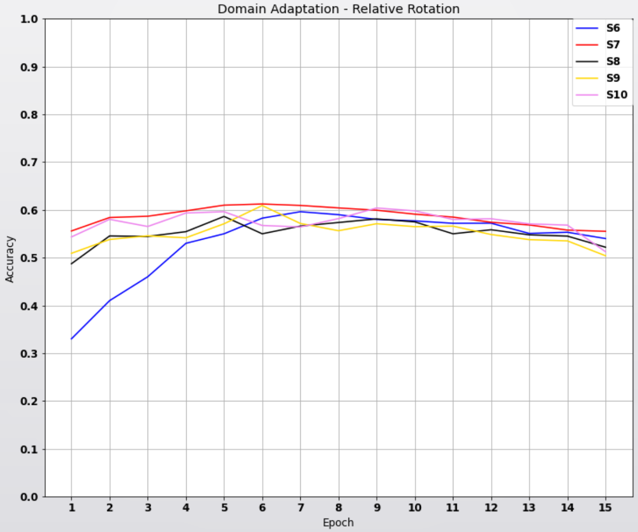
Param. Set	Batch size	LR	Weight loss Pretext task	Weight loss Entropy	weight_decay	dropout	Max. Validation Accuracy (%)
S11	32	0.0001	1	0.1	0.05	0.5	**62.92 @Epoch 3**
S12	32	0.0001	1	0.1	0	0.5	60.52 @Epoch 4
S13	32	0.0001	0.8	0.1	0.05	0.5	61.78 @Epoch 3
S14	32	0.0001	0.9	0.2	0.05	0.5	62.8 @Epoch 3
S15	32	0.0001	1.5	0.1	0.05	0.5	60.4 @Epoch 3

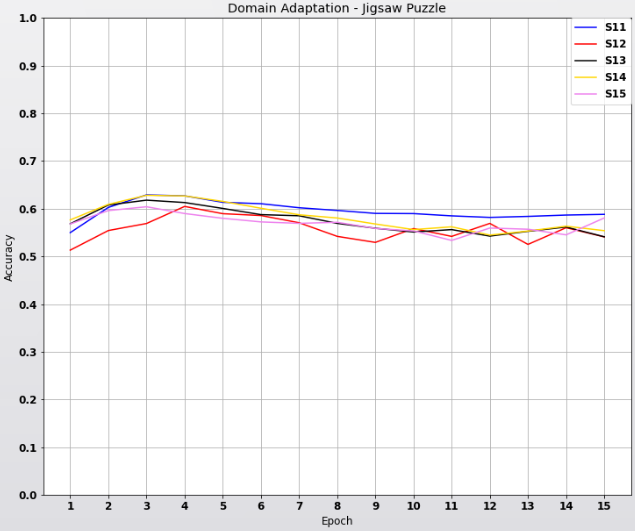


Hyperparameter tuning:

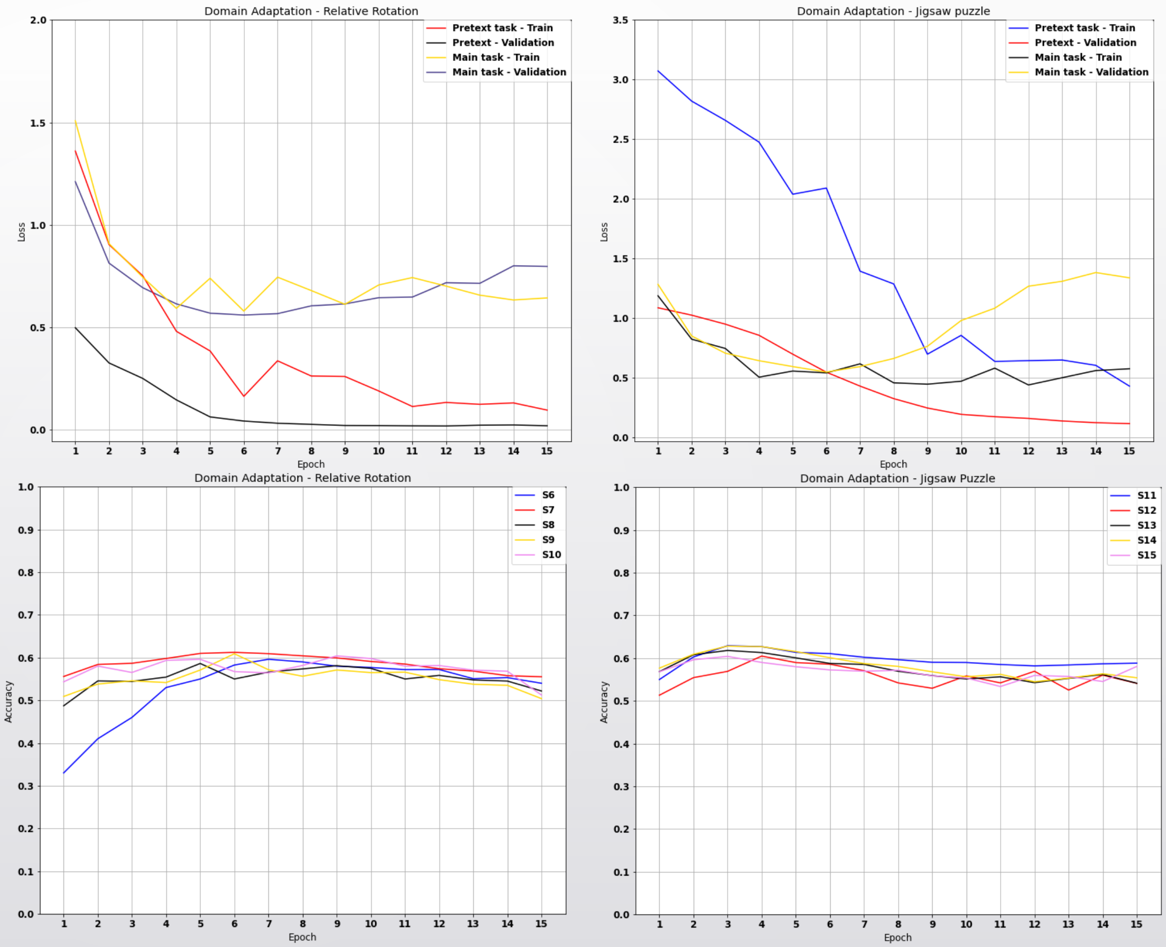
Results Comparison:







Results Comments:



Technical Difficulties

- Long time required to complete a run
- 16 GB RAM + 8 GB CUDA Memory
- CUDA toolkit versioning issues

Scores & Conclusion:

Accuracies

Experiment	Our results	Ref. results*
Source only Baseline	47.78%	47.33%
DA - Relative Rotation	61.12%	66.68%
DA - Jigsaw Puzzle	62.92%	N/A

Main experiments → overall successful:

- Comparable baseline
- DA improved the baseline score

Variations → overall successful, need further investigation:

- Acceptable scores
- Must Continue experiments

Further required investigations:

- Repeat all experiments entirely multiple times
- More fundamental approach for improving overfitting
- Perform the Jigsaw Puzzle task with 3x3(or possibly 4x4) grid

Thank you for your attention!