Improving Spatiotemporal Self-Supervision by Deep Reinforcement Learning

Uta Büchler* Biagio Brattoli* Björn Ommer

Heidelberg University, HCI/IWR, Germany {firstname.lastname}@iwr.uni-heidelberg.de

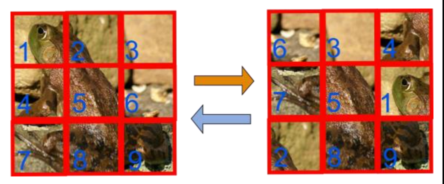
RUPRECHT-KARLS-UNIVERSITÄT HEIDELBERG · IWR Interdisciplinary Center for Scientific Computing · Computer Vision group heidelberg · Heidelberg Collaboratory for Image Processing · ECCV 2018 European Conference on Computer Vision, 8 – 14 September 2018 | Munich, Germany

1 Introduction

Self-Supervision: Learn powerful features for visual understanding utilizing large amount of unlabeled data

Surrogate Task: Ordering permuted input data

- Widely applicable (e.g. images and videos)
- Utilized in several approaches



Multi-Task Self-Supervision: Train several tasks

- Could improve upon single task training
- Problematic to balance heterogeneous tasks

Our Model:

- Unify ordering approaches in a single model
- Trained with spatial and temporal information

one network, multiple tasks, two domains } Spatiotemporal Self-Supervision

---

Procedure in Self-Supervision:

(1) apply a transformation to the input
(2) train network learning to compensate the trafo

- Tranformations are controlled by a free parameter a (e.g. selected permutation)
- Common procedure: selecting a randomly

Limitation: Does not maximize the network improvements (e.g. a problematic transformation should be shown more often than a learnt one)

Idea: Learn a policy for optimally selecting a based on the state of the network for efficient training

Random Policy	Our Policy
Independent by the model	Adjusts to the state of the net
Static policy	Dynamic policy
Inefficient Training	Efficient supervision signal
Fixed permutation distribution	Learnable distribution

Our Approach: Train a policy network using Reinforcement learning for proposing a

(A) Policy π — Permutation Proposal Network

$\pi(a|s,\theta) = P(a_t = a|s_t = s, \theta_t = \theta)$

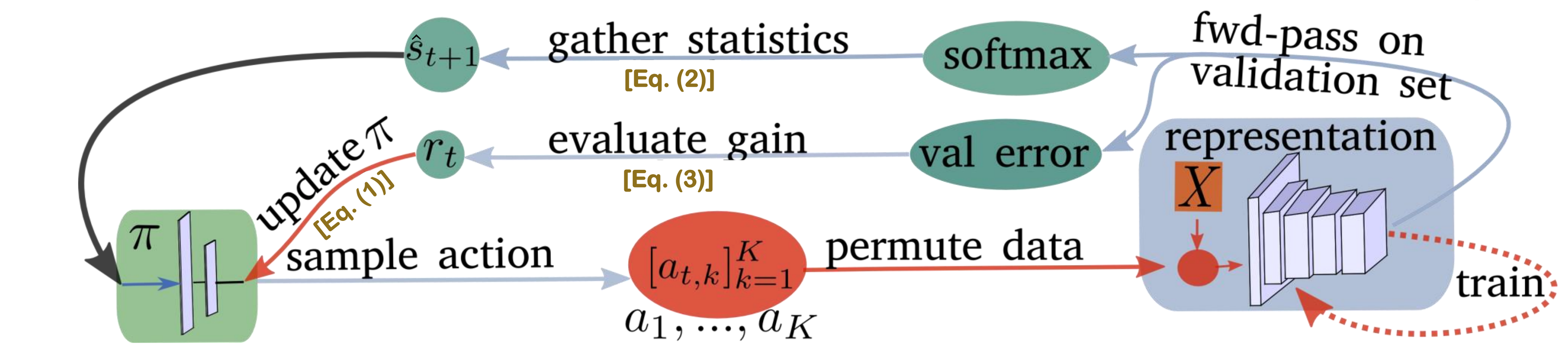
$a = (x, \psi_i)$: action

s : state of the spatiotemporal network

RL: Policy Gradient Update Rule

(1) $\theta_{t+1} = \theta_t + \alpha\left(\sum_{t'\geq t}\gamma^{t'-t}r_{t'}\right)\nabla \log \pi(a|s,\theta)$

α : learning rate
γ : discount factor
θ : weights of π



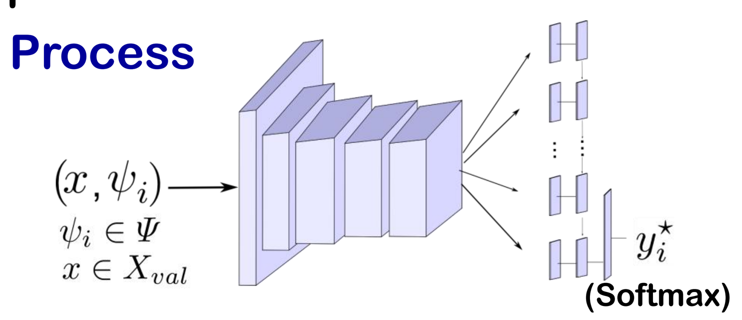
Training procedure of π

(B) Validation—State & Reward

Spatiotemporal-Network state representation

- Direct: weights → not feasible (dimensionality)
- Indirect: performance over a validation set X_{val}

Validation Process



$y_i^\star$: indicates the ability of ST-net to reconstruct ψ_i

$$s = \begin{bmatrix} y_1(x_1) & \ldots & y_1(x_{|X_{val}|}) \\ \vdots & & \vdots \\ y_{|\Psi|}(x_1) & \ldots & y_{|\Psi|}(x_{|X_{val}|}) \end{bmatrix}$$ Network State Embedding

Action Space & State Representation

Policy selects permutations directly

Issue: Complexity too high, no convergence

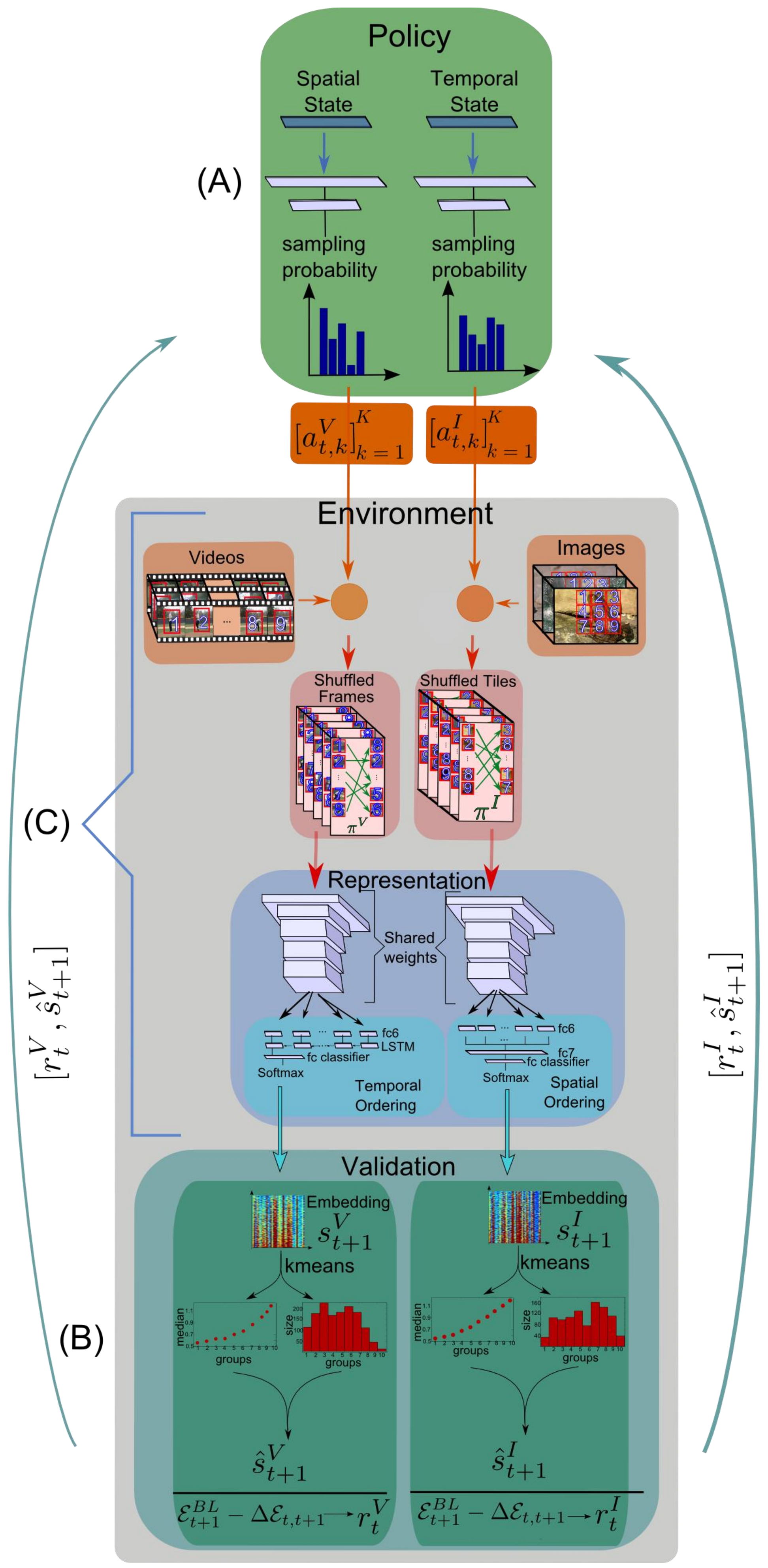
Solution: Group permutations based on s

Final State: (2) $\hat{s} = [|c_j|, median\left([s_i]_{\psi_i \in c_j}\right)]_{j=1}^{|C|}$

Reward

(3) $r_t := \mathcal{E}_{t+1}^{BL} - \mathcal{E}_{t+1}$

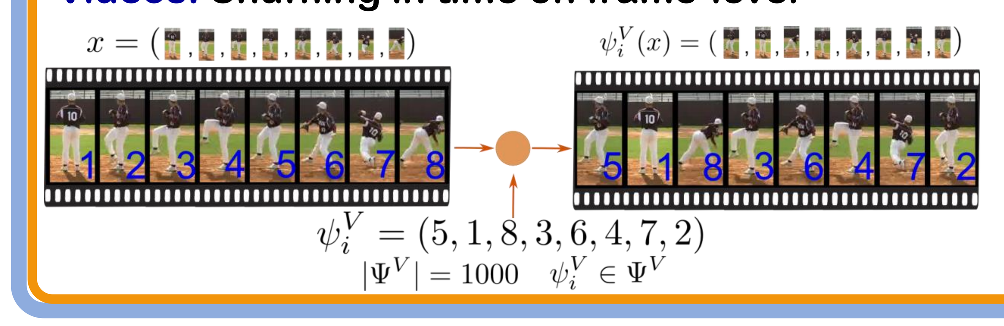
$c_j \in C$: Groups
$\mathcal{E}$: Validation Error
$\mathcal{E}_{t+1}^{BL} = 2\mathcal{E}_t - \mathcal{E}_{t-1}$: Baseline



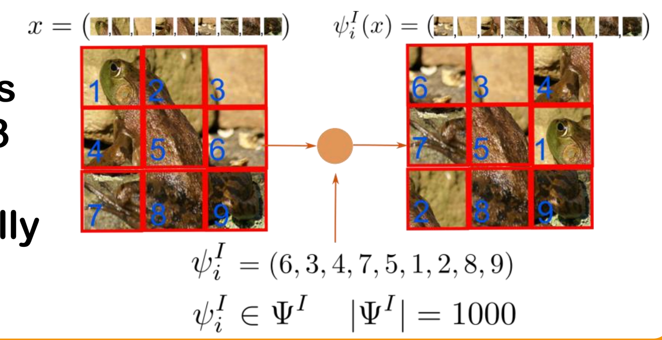
(C) Spatiotemporal Training

Data

Videos: Shuffling in time on frame-level



Images: Input is divided in a 3x3 grid of tiles shuffled spatially



Architecture

- shared CaffeNet weights until pool5
- Temporal ordering FC6—LSTM—Classifier
- Spatial ordering FC6—FC7—Classifier

2 Results—Supervised Fine-Tuning

The network is initalized with the methods shown in the first column and afterwards fine-tuned on diverse tasks.

Method	UCF-101	HMDB-51
Random	47.8	16.3
Imagenet	67.7	28.0
Shuffle&Learn [32]	50.2	18.1
VGAN [50]	52.1	-
Luo et. al [30]	53.0	-
OPN [28]	56.3	22.1
Jigsaw* [34]	51.5	22.5
Ours	**58.6**	**25.0**

Action Recognition

Method	Non-Linear	Linear
Imagenet	59.7	50.5
Random	12.0	14.1
RotNet+[18]	43.8	36.5
Videos [52]	29.8	-
OPN* [28]	29.6	-
Context [10]	30.4	29.6
Colorization[55]	35.2	30.3
BiGan[12]	34.8	28.0
Split-Brain[56]	-	32.8
NAT[5]	36.0	-
Jigsaw[34]	34.6	27.1
Ours	**38.2**	**36.5**

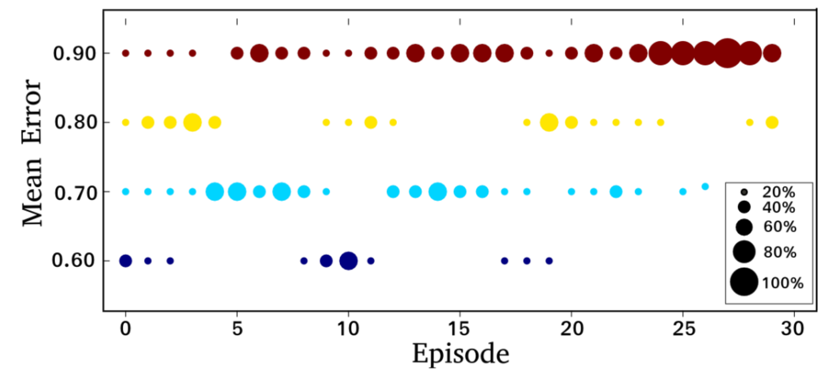
Imagenet Classification

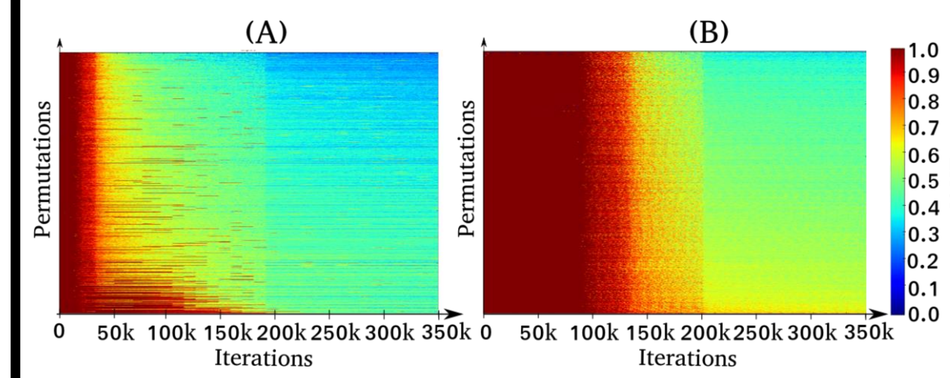
Method	Classification[13]	Detection[13]	Segmentation[14]
Imagenet	78.2	56.8	48.0
Random	53.3	43.4	19.8
RotNet[18]+	73.0	54.4	39.1
OPN[28]	63.8	46.9	-
Color17[27]	65.9	-	38.4
Counting[35]	67.7	51.4	36.6
PermNet[9]	69.4	49.5	37.9
Jigsaw[34]	67.6	**53.2**	37.6
Ours	**74.2**	52.8	**42.8**

Pascal VOC Dataset

3 Ablation Studies

Permutations (grouped by $\mathcal{E}$) chosen by the policy in each training step. Policy learns to sample **hard permutations in later itations.**





$\mathcal{E}$ over time, one permutation per row. The policy leads to **faster progress (A)** upon training with random policy which improves uniformly (B)

Method	S	S+P	T	T+P	S&T	S+T	S+T+P
Pascal	67.6	71.3	64.1	65.9	69.8	72.0	**74.2**
UCF-101	51.5	54.6	52.8	55.7	54.2	57.3	**58.6**

Supervised fine-tuning for action recognition and multi-object classification. **Joint training** (S+T) **outperforms** serial training(S&T). Utilizing P improves upon the random policy.