ICLR 2024

CONTINUAL MOMENTUM FILTERING ON PARAMETER SPACE FOR ONLINE TEST-TIME ADAPTATION

Jae-Hong Lee ·
Joon-Hyuk Chang
Hanyang University

한양대학교
HANYANG UNIVERSITY

Unsupervised online domain adaptation

Online test-time adaptation (OTTA)

- Challenges of OTTA
  - Unsupervised domain adaptation → Models are trained in an unsupervised manner, thus cannot utilize ground truth labels during training.
  - Source-free → Does not allow access to source data, only permits the use of the source model.
  - Online learning → Allows only a one-time access to target domain samples.
  - By resolving such challenging issues, it is possible to perform interaction adaptation.
- Various scenarios of OTTA
  - covariate shifts (CS),
  - temporally-correlated covariate shifts (TC-CS),
  - temporally-correlated label shifts (TC-LS) over CS
  - TC-CS over TC-CS
- Applications → self-driving, speech recognition, personalization, smart factory, etc.

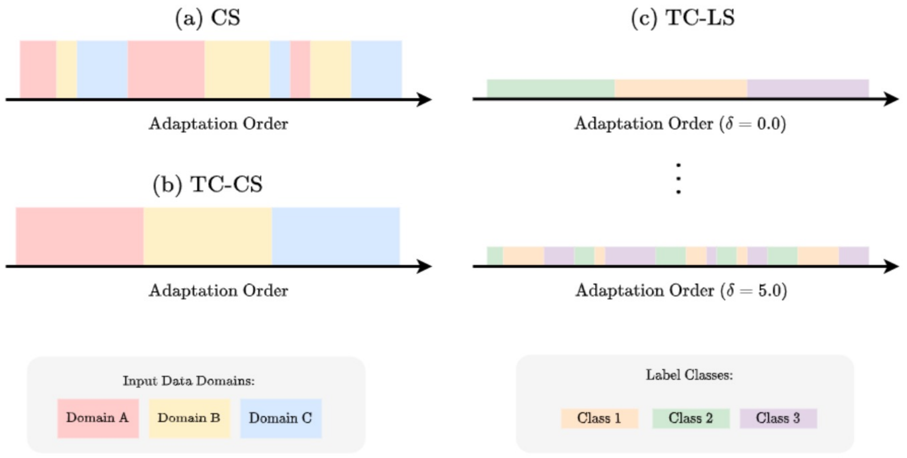


Figure. Various scenarios of OTTA

Introduction

Problem

Non-independent and identically distributed samples

- DNN models rely on the i.i.d assumption.
- The i.i.d assumption is difficult to maintain in stream data for domain adaptation.
  - (non-independent) Data obtained from nature are temporally correlated.
  - (non-identical) Shifts occur between the source and target distributions.
- If the assumption is not met, the performance of DNNs drops significantly.

Error propagation from catastrophic forgetting

- Catastrophic forgetting
  - When distribution shifts occur, the performance of the source model decreases.
  - In non-independent sampling situations, specific biases are introduced to the model (e.g., mode collapse), and catastrophic forgetting occurs.
- Error propagation
  - If self-training is conducted using unreliable model outputs, error propagation is accelerated and catastrophic forgetting occurs.
- Catastrophic forgetting and error propagation form a negative feedback loop.

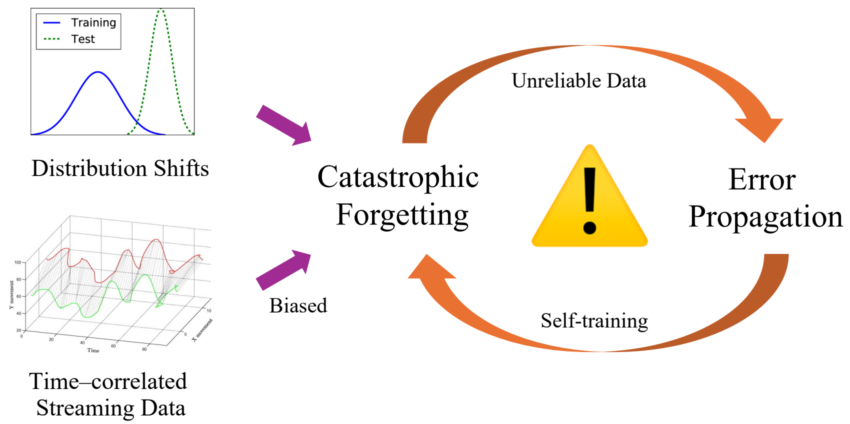


Figure. Accelerated catastrophic forgetting

Source model dependency

Introduction

- Using noisy predictions as labels to train, the parameters of the target model become noisy due to error propagation.
  - Method of training only a subset of source model parameters.
  - Constraining the target model with fixed information or parameters from the source model

    → **By regularizing to prevent the target model from diverging too far from the source model**, catastrophic forgetting is prevented.
- Existing methods limit the flexibility of the target model because they continuously use the information from **the frozen source model.**
  - It is difficult **to adapt to distribution shifts in the target domain.**

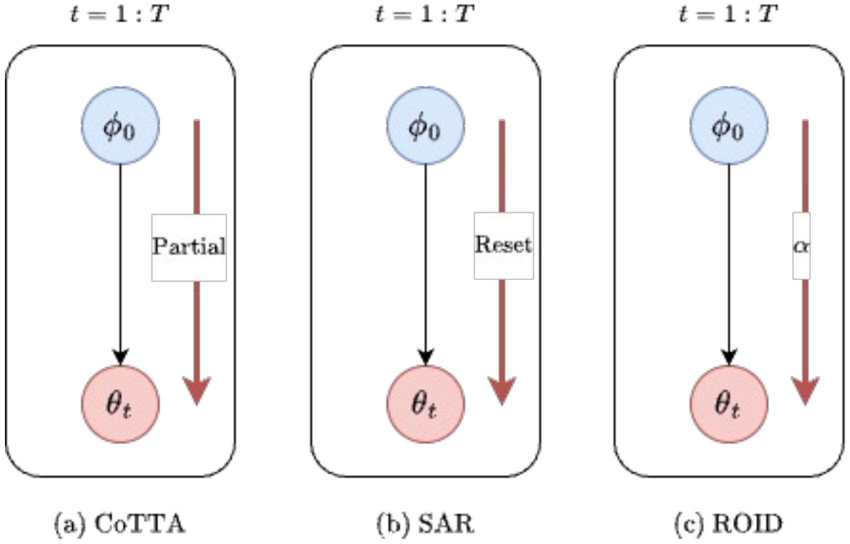


Figure. Graphical model of exist OTTA methods

Methodology

Continual momentum filtering on parameter space for online test-time adaptation

Motivation

- Use a hidden model instead of freezing the source model
  - The hidden model updates with target model parameters
  - Increased risk of error propagation due to noisy target model
- Adopt Kalman filtering with noise reduction capabilities, applied in the parameter space
  - Kalman filtering models the intrinsic noise of observations
  - Observations are set as target model parameters
  - Kalman filtering suppresses noisy observations, which are then stored in the hidden model
- Overall Framework
  - Optimization process based on Stochastic Gradient Descent (SGD)
  - Inference process based on Kalman filtering
  - Alternating between the two processes, performing the OTTA procedure

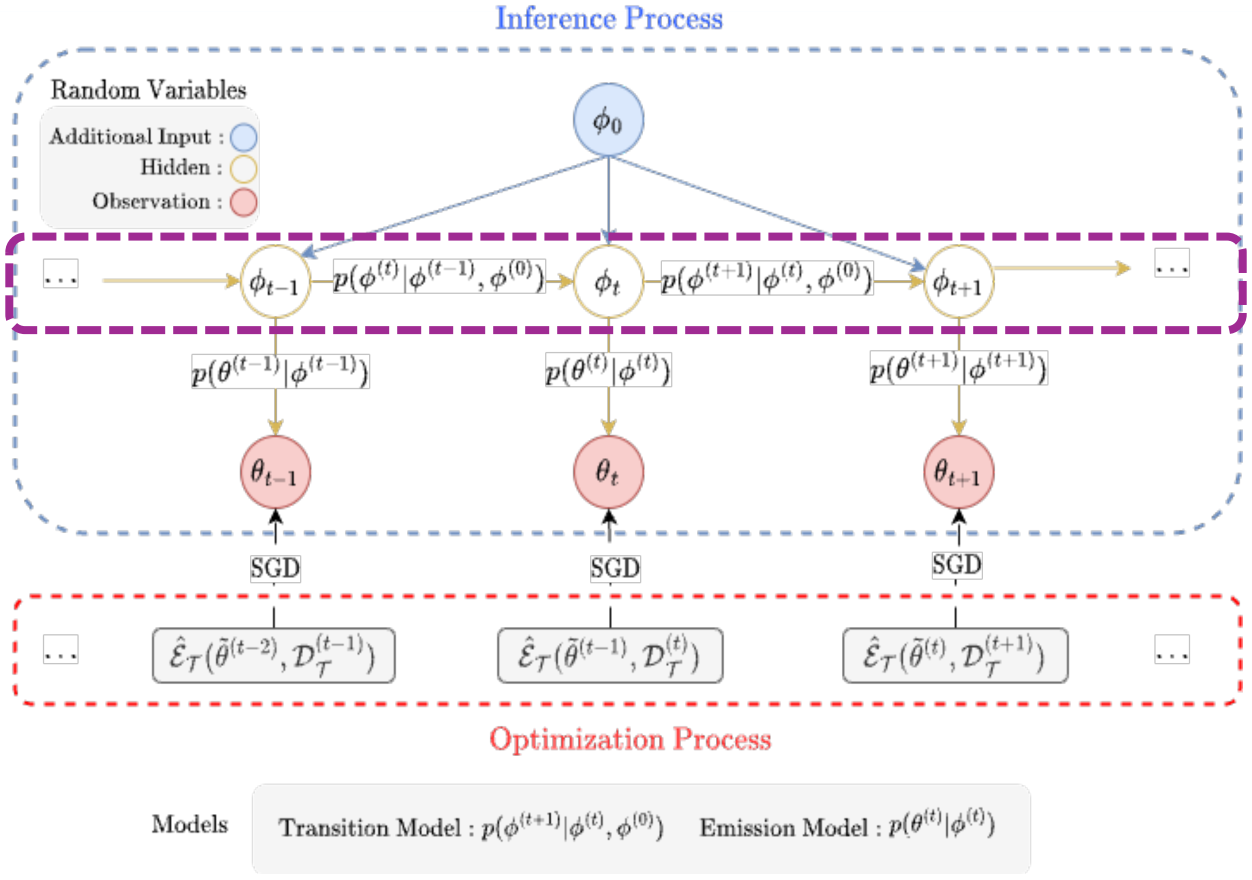


Figure. Graphical model of continual momentum filtering (CMF)

Continual momentum filtering on parameter space for online test-time adaptation

Optimization process

- Optimization process (generic)

$$\hat{\mathcal{E}}_{\mathcal{T}}(\Theta^{(0)}, \mathcal{D}_{\mathcal{T}}^{(t)}) = \frac{1}{N_{\mathcal{T}}} \sum_{\mathbf{x}_n \in \mathcal{D}_{\mathcal{T}}^{(t)}} \ell(f(\mathbf{x}_n; \Theta^{(0)})).$$

$$\Theta^{(t+1)} = \arg\min_{\Theta^{(t)}} \hat{\mathcal{E}}_{\mathcal{T}}(\Theta^{(t)}, \mathcal{D}_{\mathcal{T}}^{(t+1)}) + \cancel{\lambda \mathrm{d}(\Theta^{(0)}, \Theta^{(t)})},$$

Regularization term

- Optimization process (in CMF)

$$\theta^{(t+1)} = \arg\min_{\tilde{\theta}^{(t)}} \hat{\mathcal{E}}_{\mathcal{T}}(\tilde{\theta}^{(t)}, \mathcal{D}_{\mathcal{T}}^{(t+1)}).$$

Calculate by CMF
in ference process

  - Remove the regularization term composed of source parameters.
  - The refined parameter $\hat{\theta}^{(t)}$ calculated by CMF is used for regularization of the hidden model parameters.

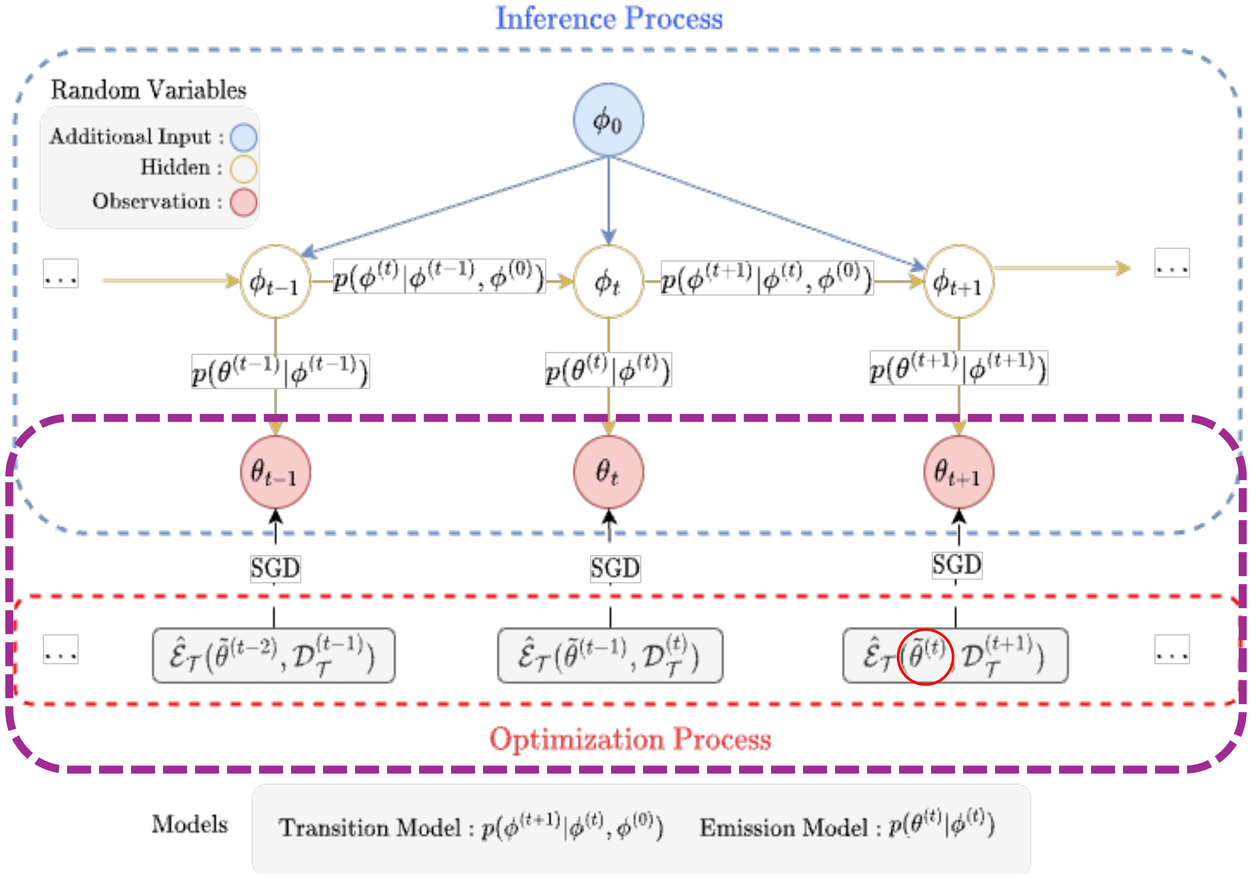


Figure. Graphical model of continual momentum filtering (CMF)

Continual momentum filtering on parameter space for online test–time adaptation

Parameterization

- Transition model

$$p(\phi^{(t)}|\phi^{(t-1)},\phi^{(0)}) = \mathcal{N}(\phi^{(t)}|\mathrm{A}\phi^{(t-1)} + (1-\mathrm{A})\phi^{(0)}, Q),$$

  - Design the transition model using the source parameter as an auxiliary variable.
  - The role is to recover the hidden parameter that can be distorted when updated with target parameters.

- Emission model

$$p(\theta^{(t)}|\phi^{(t)}) = \mathcal{N}(\theta^{(t)}|\mathrm{H}\phi^{(t)}, R),$$

  - Assume that there will be little change in observations since it targets a well–pretrained source model.

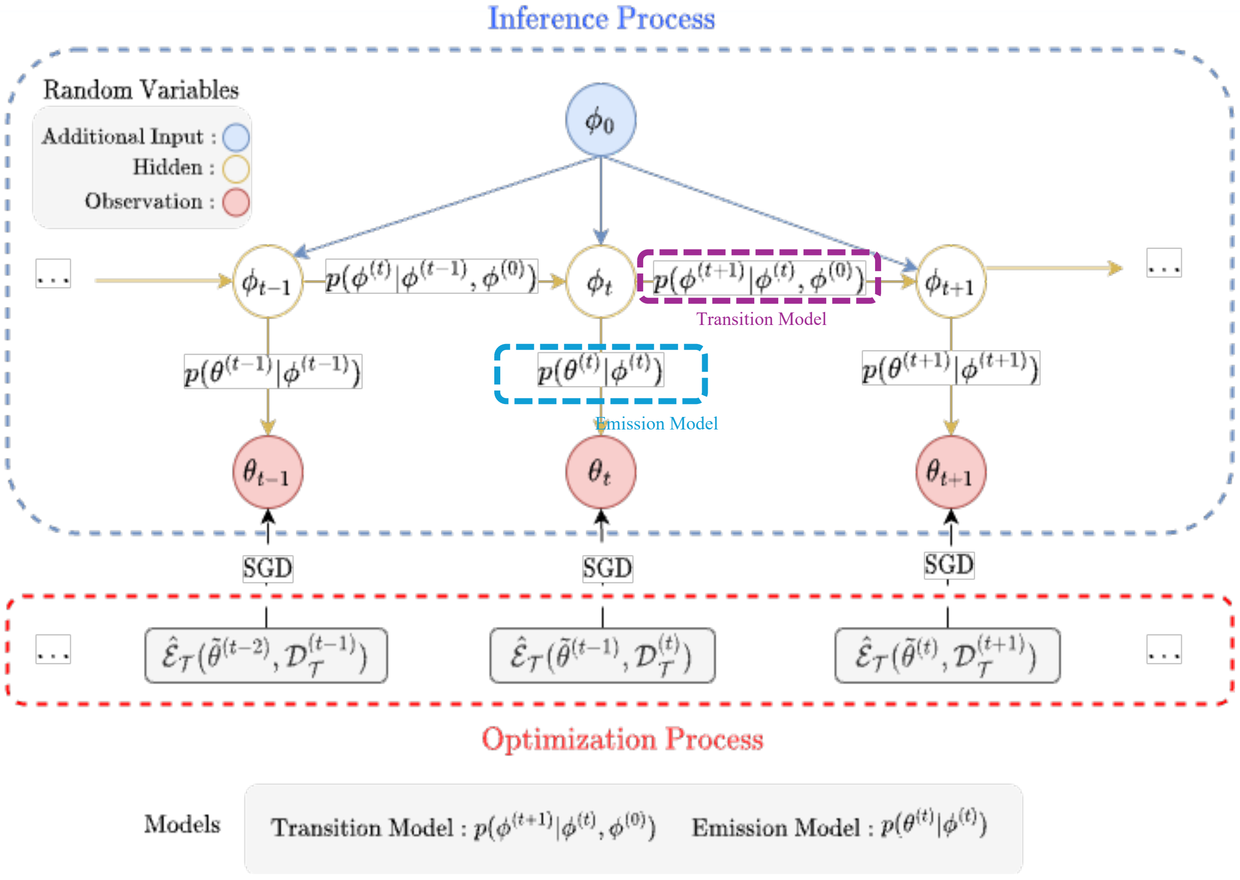


Figure. Graphical model of continual momentum filtering (CMF)

Continual momentum filtering on parameter space for online test-time adaptation

Inference process

- Predict step

$$p(\phi^{(t)}|\theta^{(1:t-1)},\phi^{(0)}) = \mathcal{N}(\phi^{(t)}|\mu_{t|t-1},\Sigma_{t|t-1})$$
$$\mu_{t|t-1} = \mathrm{A}\mu_{t-1|t-1} + (1-\mathrm{A})\phi^{(0)},$$
$$\Sigma_{t|t-1} = \mathrm{A}\Sigma_{t-1|t-1}\mathrm{A}^\top + Q.$$

- Update step

$$p(\phi^{(t)}|\theta^{(1:t)},\phi^{(0)}) = \mathcal{N}(\phi^{(t)}|\mu_{t|t},\Sigma_{t|t}),$$
$$\mathrm{K}_t = \Sigma_{t|t-1}\mathrm{H}^\top(\mathrm{H}\Sigma_{t|t-1}\mathrm{H}^\top + R)^{-1},$$
$$\mu_{t|t} = \mu_{t|t-1} + \mathrm{K}_t(\theta^{(t)} - \mathrm{H}\mu_{t|t-1}),$$
$$\Sigma_{t|t} = \Sigma_{t|t-1} - \mathrm{K}_t\mathrm{H}\Sigma_{t|t-1}.$$

- Transfer step

$$\tilde{\theta}^{(t)} = \Gamma\theta^{(t)} + (1-\Gamma)\mu_{t|t},$$

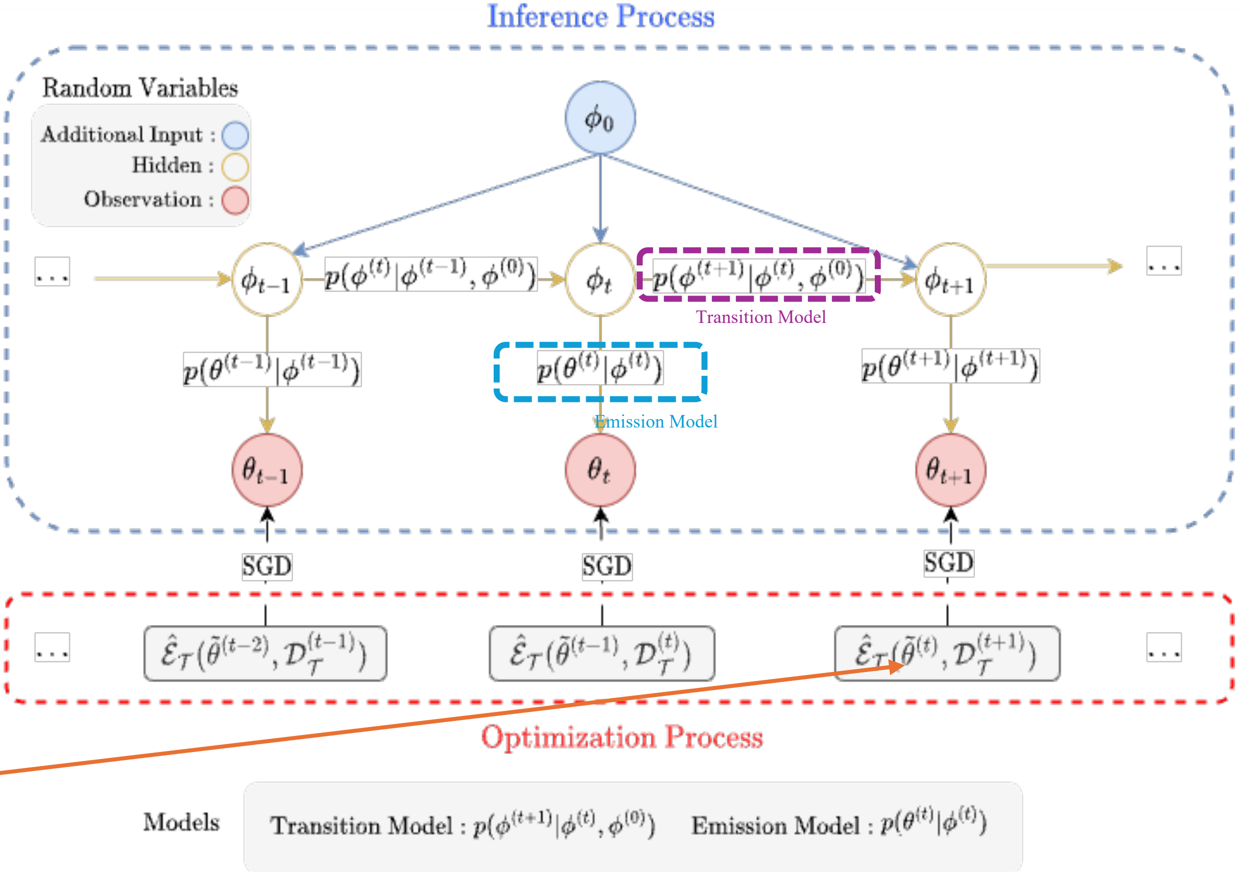


Figure. Graphical model of continual momentum filtering (CMF)

Methodology

Continual momentum filtering on parameter space for online test-time adaptation

Stable Implementation

- The value of the parameter dimension of DNNs is generally large, which requires high compute cost for Kalman filtering.
- Simplify the Kalman filtering parameter to a scalar.

$$(\mathrm{A}, Q, \mathrm{H}, R, \Gamma) \xrightarrow[\text{Scalarization}]{} (\alpha, q, \eta, r, \gamma)$$

- In the simplified setting, CMF is simplified as follows:

$$\boxed{\begin{aligned} \mu_{t|t-1} &= \text{Moments}(\mu_{t-1|t-1}, \phi^{(0)}, \alpha), \\ \mu_{t|t} &= \text{Moments}(\mu_{t|t-1}, \theta^{(t)}, \beta_t), \\ \tilde{\theta}^{(t)} &= \text{Moments}(\theta^{(t)}, \mu_{t|t}, \gamma), \\ \text{Moments}(x_1, x_2, a) &= a x_1 + (1-a) x_2, \end{aligned}}$$

$$\boxed{\begin{aligned} \Sigma_{t|t-1} &= \alpha^2 \Sigma_{t-1|t-1} + q, \\ \beta_t &= r/(\Sigma_{t|t-1} + r), \\ \Sigma_{t|t} &= \beta_t \Sigma_{t|t-1}. \end{aligned}}$$

Algorithm 1 Continual Momentum Filtering

INPUT:
Input data stream $\{\mathcal{D}_T^{(1)} \dots \mathcal{D}_T^{(T)}\}$, Source model $f(.;\theta^0)$, Number of updates I,
Hyperparameter (α, q, r, γ), Initialization $\tilde{\theta}^{(0)} \leftarrow \theta^{(0)}$, $\mu_{0|0} \leftarrow \theta^{(0)}$, $\Sigma_{0|0} \leftarrow 0$
for $t = 1, \dots, T$ **do**
  for $i = 1, \dots, I$ **do**
    OPTIMIZATION PROCESS:
      $\theta^{(t)} = \arg\min_{\tilde{\theta}^{(t-1)}} \hat{\mathcal{E}}_T(\tilde{\theta}^{(t-1)}, \mathcal{D}_T^{(t)})$ ▷ Eq. (4)
    INFERENCE PROCESS:
      // Predict Step:
      $\mu_{t|t-1} = \text{Moments}(\mu_{t-1|t-1}, \phi^{(0)}, \alpha)$ ▷ Eq. (15)
      $\Sigma_{t|t-1} = \alpha^2 \Sigma_{t-1|t-1} + q$ ▷ Eq. (18)
      // Update Step:
      $\beta_t = r/(\Sigma_{t|t-1} + r)$ ▷ Eq. (19)
      $\mu_{t|t} = \text{Moments}(\mu_{t|t-1}, \theta^{(t)}, \beta_t)$ ▷ Eq. (16)
      $\Sigma_{t|t} = \beta_t \Sigma_{t|t-1}$ ▷ Eq. (20)
      // Parameter Ensemble:
      $\tilde{\theta}^{(t)} = \text{Moments}(\theta^{(t)}, \mu_{t|t}, \gamma)$ ▷ Eq. (17)
  end for
end for

Continual momentum filtering on parameter space for online test-time adaptation

Experimental Settings (Image)

- Dataset
  - Source data
    - → ImageNet-1K
  - Target Data
    - → ImageNet-C
    - → ImageNet-D109 (D109)
    - → ImageNet-R (Rendition)
    - → ImageNet-Sketch (Sketch)
- Models
  - VisionTransformer (ViT), SwinTransformer (Swin), data2vec-vision (D2V)
- Comparison Methods
  - TENT, CoTTA, RoTTA, SAR, EATA, ROID
- Performance Metric
  - 4 random seeds
  - Average error rates (%)

Experimental Settings (Speech)

- Dataset
  - Source data
    - → LibriSpeech (LS)
    - → LbriVox (Vox)
  - Target Data
    - → TED-LIUM v3 (TED)
    - → Common Voice (CV)
- Models
  - data2vec base (D2V-Libri), data2vec large (D2V-Vox)
- Comparison Methods
  - SUTA (continual, episodic)
- Performance Metric
  - 4 random seeds
  - Viterbi Decoding
  - Word Error Rate (WER) (%)

[8] Lee, Jae-Hong, and Joon-Hyuk Chang. "Continual Momentum Filtering on Parameter Space for Online Test-time Adaptation." The Twelfth International Conference on Learning Representations. 2023.

Continual momentum filtering on parameter space for online test-time adaptation

Experimental Results

- TENT experiences performance degradation compared to the source model in both ImageNet-C and D109 datasets, SAR in D109, and EATA in ImageNet-C.
- RoTTA, CoTTA, and ROID show relatively robust performance, with ROID having the highest performance among them.
- CMF achieves the lowest mean error rates among the existing methods, consistently showing performance improvements across all models.

Method	ImageNet-C				D109			
	ResNet-50	ViT	Swin	D2V	ResNet-50	ViT	Swin	D2V
Source	82.0	60.2	64.0	51.8	58.8	53.6	51.4	48.0
TENT	85.7±0.95	55.1±0.08	62.6±0.18	50.5±0.06	55.4±0.08	76.8±0.36	61.5±0.41	57.9±0.42
CoTTA	82.0±0.08	59.6±0.02	63.9±0.01	51.2±0.02	55.3±0.04	53.3±0.04	51.2±0.03	47.8±0.01
RoTTA	79.5±0.10	58.7±0.04	62.9±0.03	51.3±0.03	54.8±0.04	50.9±0.05	48.6±0.05	46.8±0.03
SAR	79.6±0.68	52.3±0.12	60.5±1.04	50.7±0.07	53.6±0.07	61.2±0.36	53.9±0.08	48.1±0.08
EATA	72.5±1.44	51.8±0.14	56.2±0.29	76.2±20.23	53.1±0.09	48.5±0.11	48.8±0.12	46.2±0.05
ROID	69.5±0.13	50.7±0.08	55.0±0.26	47.4±0.08	50.9±0.04	46.9±0.02	47.2±0.07	45.0±0.01
CMF (ours)	**67.6±0.20**	**49.0±0.10**	**52.1±0.12**	**45.7±0.03**	**49.4±0.21**	**44.5±0.08**	**44.8±0.04**	**42.8±0.05**

Table. Average error rates (%) and their corresponding standard deviations in the scenario of CS. Red fonts indicate performance degradation.

Continual momentum filtering on parameter space for online test-time adaptation

Experimental Results

- TENT suffers severe performance degradation in the CS scenario, and even CoTTA and RoTTA, which were robust, experience a decline.
- EATA shows robust performance except for the D2V model but does not match ROID.
- CMF achieves the lowest average error rates among existing methods in this scenario as well. CMF consistently demonstrates performance improvements across various datasets and models.

Method	ImageNet-C			D109			Rendition			Sketch		
	ViT	Swin	D2V	ViT	Swin	D2V	ViT	Swin	D2V	ViT	Swin	D2V
Source	60.2	64.0	51.8	53.6	51.4	48.0	56.0	54.2	46.6	70.6	68.4	60.4
TENT	54.5±0.04	64.0±0.14	51.9±0.09	83.3±0.13	66.4±0.33	62.9±0.21	53.3±0.09	53.8±0.38	46.0±0.03	70.8±1.12	68.7±0.22	60.3±0.06
CoTTA	60.4±0.02	64.2±0.01	51.7±0.02	53.3±0.03	51.2±0.01	47.8±0.02	55.6±0.03	54.1±0.02	46.4±0.01	70.6±0.01	68.3±0.02	60.3±0.01
RoTTA	59.1±0.05	63.4±0.01	51.3±0.01	51.4±0.03	49.1±0.03	47.2±0.03	54.8±0.04	53.5±0.03	46.5±0.02	69.3±0.03	67.3±0.03	60.1±0.03
SAR	51.7±0.14	65.9±1.27	51.0±0.12	57.3±0.41	53.5±1.05	48.5±0.10	48.5±0.21	53.7±2.78	45.9±0.05	70.5±1.21	73.4±1.31	60.2±0.07
EATA	49.9±0.06	52.9±0.25	64.4±15.84	47.2±0.10	47.4±0.18	45.8±0.06	49.0±0.20	49.9±0.33	45.0±0.08	59.8±0.19	60.6±0.26	78.3±17.08
ROID	45.0±0.09	47.0±0.26	44.8±0.01	45.0±0.04	45.1±0.10	44.2±0.06	44.2±0.13	46.0±0.10	41.8±0.11	58.6±0.04	58.9±0.11	56.2±0.05
CMF (ours)	**44.8±0.12**	**46.6±0.12**	**43.5±0.04**	**43.4±0.07**	**43.6±0.12**	**42.3±0.11**	**42.7±0.20**	**44.1±0.24**	**40.0±0.06**	**57.0±0.08**	**56.7±0.13**	**53.9±0.03**

Table. Average error rates (%) and their corresponding standard deviations in the scenario of TC- CS. Red fonts indicate performance degradation with respect to Source.

Continual momentum filtering on parameter space for online test-time adaptation

Experimental Results

- Firstly, in the case of the highest degree of temporal correlation (i.e., δ=0.0), all methods except for LAME and ROID show unstable results.
- Among the two methods, ROID shows the highest performance in all models except for the Swin model in D109 → CMF shows lower error rates than both methods across all three models.
- As the temporal correlation decreases, LAME experiences a severe performance drop.
- Meanwhile, SAR and EATA show relatively competitive performance but do not reach the level of ROID → CMF outperforms ROID in all cases.

δ	Model	ImageNet-C					D109				
		LAME	SAR	EATA	ROID	CMF (ours)	LAME	SAR	EATA	ROID	CMF (ours)
0.0	ViT	44.1±0.02	48.3±0.28	71.8±1.22	16.2±0.06	**15.9±0.04**	35.2±0.55	58.5±0.40	58.6±1.45	31.4±0.07	**31.0±0.10**
	Swin	47.1±0.09	60.1±0.74	72.7±0.67	18.1±0.03	**16.7±0.10**	30.1±0.16	55.4±0.17	54.2±0.99	30.3±0.25	**29.6±0.21**
	D2V	38.9±0.07	48.3±0.15	58.2±2.21	17.4±0.21	**14.4±0.24**	29.7±0.15	49.5±0.04	46.1±0.37	29.3±0.03	**27.8±0.12**
0.01	ViT	83.2±0.23	48.7±0.29	47.7±0.12	36.3±0.08	**35.0±0.04**	44.8±0.69	58.6±0.80	50.7±1.20	32.2±0.10	**31.8±0.10**
	Swin	84.7±0.12	58.4±0.86	50.0±0.35	37.2±0.06	**35.1±0.16**	39.9±0.77	53.7±0.53	49.6±0.41	31.1±0.11	**30.3±0.24**
	D2V	79.5±0.20	47.9±0.05	65.0±18.58	35.9±0.08	**32.7±0.04**	39.9±0.56	49.1±0.14	47.1±1.08	30.7±0.09	**28.6±0.11**
0.1	ViT	79.9±0.06	48.4±0.30	46.1±0.17	41.3±0.05	**39.6±0.03**	68.9±0.24	57.7±0.56	47.4±0.16	37.3±0.12	**36.1±0.11**
	Swin	84.5±0.09	58.4±0.75	48.3±0.09	42.1±0.04	**39.6±0.02**	64.6±0.25	53.4±0.70	47.4±0.21	36.9±0.11	**35.0±0.05**
	D2V	70.1±0.04	48.0±0.04	65.5±19.11	41.3±0.03	**38.2±0.05**	64.6±0.25	48.6±0.04	45.7±0.08	36.3±0.06	**34.1±0.13**
1.0	ViT	80.0±0.03	48.3±0.25	45.7±0.15	41.2±0.03	**39.4±0.03**	90.0±0.09	57.4±0.12	47.2±0.04	42.9±0.03	**41.3±0.06**
	Swin	84.6±0.06	58.5±0.41	47.4±0.39	41.9±0.03	**39.4±0.11**	86.9±0.24	54.5±0.68	47.4±0.10	43.0±0.06	**41.3±0.04**
	D2V	70.2±0.07	47.9±0.09	87.0±18.44	41.2±0.01	**38.1±0.03**	88.3±0.13	48.5±0.09	45.7±0.04	42.2±0.04	**40.1±0.10**
5.0	ViT	80.2±0.09	55.5±12.62	45.6±0.17	41.3±0.03	**39.5±0.03**	93.3±0.17	57.3±0.22	47.2±0.08	43.9±0.09	**42.5±0.08**
	Swin	84.9±0.04	59.2±0.68	47.6±0.25	41.9±0.03	**39.4±0.08**	90.6±0.23	54.0±0.72	47.3±0.05	44.1±0.06	**42.5±0.07**
	D2V	70.5±0.12	47.9±0.08	65.9±18.92	41.2±0.03	**38.0±0.05**	92.8±0.16	48.4±0.12	45.7±0.06	43.2±0.04	**41.1±0.06**

Table. Average error rates (%) and their corresponding standard deviations in the scenario of TC- LS over TC-CS.

Continual momentum filtering on parameter space for online test-time adaptation

Experimental Results

- Complex distribution shifts
  - For ImageNet-C and D109, δ was experimented with at 0.01 and 0.1 respectively.
  - CMF shows the best performance across all datasets and models.
- Real-world streaming
  - Both TED and CV differ from the source domain LibriSpeech in terms of recording environment and the domain of words used.
  - SUTA is an episodic method that performs test-time adaptation for a single utterance.
  - When applied to a continual setting, there is a severe performance degradation.
  - CMF prevents catastrophic forgetting of SUTA in continual settings and improves performance compared to episodic SUTA.

Method	ImageNet-C			D109		
	ViT	Swin	D2V	ViT	Swin	D2V
LAME	36.1±0.09	37.4±0.12	36.3±0.11	29.9±0.18	28.6±0.23	29.1±0.19
SAR	54.1±0.40	65.4±0.53	47.2±0.08	61.0±0.51	53.6±0.24	48.6±0.35
EATA	70.5±0.67	77.1±0.93	85.8±18.90	52.9±2.98	50.3±0.25	45.9±0.13
ROID	23.6±0.05	28.6±0.16	18.8±0.01	29.1±0.09	28.2±0.05	26.3±0.07
CMF (ours)	**23.2±0.05**	**27.1±0.08**	**17.1±0.09**	**28.7±0.19**	**27.3±0.05**	**24.9±0.10**

Table. Average error rates (%) and their corresponding standard deviations in the scenario of TC-LS over CS.

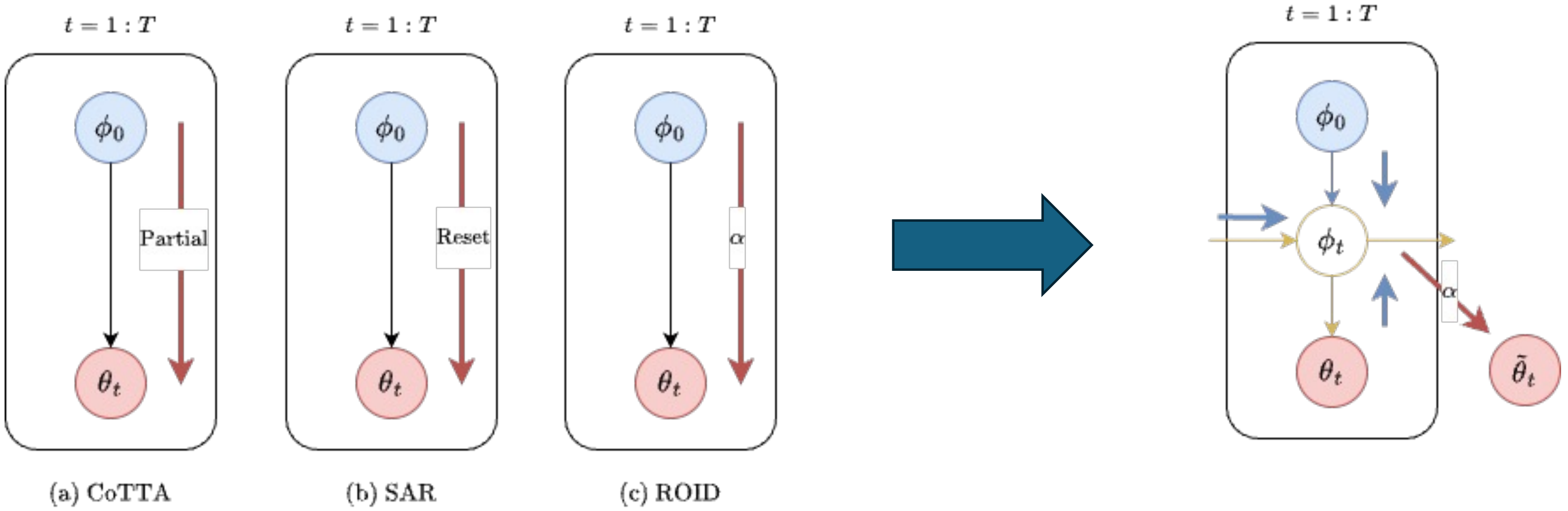
Method	TED		CV	
	D2V-Libri	D2V-VOX	D2V-Libri	D2V-VOX
Source	12.2	8.5	33.4	20.6
SUTA-cont.	67.7±1.70	66.1±0.36	120.89±4.03	130.3±1.88
SUTA-episodic	12.0±0.03	8.0±0.03	30.3±0.01	18.9±0.01
CMF (ours)	**11.8±0.05**	**7.9±0.02**	**29.6±0.02**	**18.7±0.03**

Table. Average WERs (%) and their corresponding standard deviations in real-world streaming scenario.

Continual momentum filtering on parameter space for online test-time adaptation

Conclusion

- We propose CMF, which utilizes the Kalman filter to denoise target models along with the source model, infers a new source model, and thereby refines the OTTA method.
- By simplifying the Kalman filter algorithm, we reduce computation and ensure the practicality of CMF.
- Our framework has been validated across various scenarios tested with existing OTTA methods and has shown significant performance improvements.
- It also yields valid results in the real-world streaming scenario of the speech recognition task.



[8] Lee, Jae-Hong, and Joon-Hyuk Chang. "Continual Momentum Filtering on Parameter Space for Online Test-time Adaptation." The Twelfth International Conference on Learning Representations. 2023.