A Appendix / Supplemental Material

A.1 Datasets

All datasets provide video recordings with a resolution of 640×480, and frame rate of 30 FPS. Below we provide data-specific details.

iBVP [29]: The iBVP dataset consists of 124 synchronized RGB and thermal infrared videos from 31 subjects, acquired under controlled conditions. Each video is 3 minutes in duration, and the ground truth BVP signals were acquired from the ear using PhysioKit [30]. Data were acquired under 4 different conditions that include controlled breathing, math tasks, and head movements. BVP signals are marked with the signal quality, enabling the use of the video frames only where the quality of ground-truth BVP signal is high. In this work, we use only RGB frames to train the models.

PURE [55]: This data set comprises video recordings from 10 subjects, with the ground-truth BVP and SpO2 signals acquired from the subject's finger. For each participant, six recordings are acquired under varied motion conditions, offering a range of data reflecting different physical states.

UBFC-rPPG [2]: This data set contains video recordings of 43 subjects acquired under indoor conditions with a combination of natural sunlight and artificial illumination.

SCAMPS [42]: This dataset comprises 2800 videos of synthetic avatars that were generated through high-fidelity, quasi-photorealistic renderings. Although the videos introduce various conditions such as head motions, facial expressions, and changes in ambient illumination, they are often used as a training set rather than a validation or test set.

A.2 Implementation Overview

The preprocessing steps for video frames include face detection using the YOLO5Face [49] face detector at an interval of 30 frames and using the detected facial bounding box to crop 30 subsequent frames, prior to performing the next face detection. The cropped facial frames are resized to a resolution of 72×72, which has been shown to be sufficient to estimate the rPPG. Additionally, to ensure uniform input data for all models, we add `Diff` layer to the PhysNet [83] and PhysFormer [77] architectures, as implemented by EfficientPhys [37] and the proposed FactorizePhys models, and train all the models from scratch using uniformly preprocessed video frames.

The number of frames in a video chunk is maintained as 161, which after the `Diff` layer becomes 160, making the spatial-temporal input data size $160 \times 72 \times 72$. Ground-truth BVP signals are also uniformly standardized for training all models. This is different from some of the recent work [37] that applies `Diff` in addition to standardization. We empirically found that all models perform significantly better when trained with the standardized BVP signals, although when the `Diff` is applied to the video frames.

All models were trained with 10 epochs on, following a recent work [79], as a higher number of epochs, e.g. 30 epochs as used in rPPG-Toolbox [38] resulted in poor generalization for all models. However, we used only one epoch for all models to train on the SCAMPS [42] dataset, since this dataset is a synthesized dataset with generated BVP signals that are easier for models to learn, unlike real-world datasets. Training beyond one epoch resulted in poorer cross-dataset performance for all the models. The batch size of 4 was used consistently throughout the training and the maximum learning rate was set to 1×10^{-3} with 1 cycle learning rate scheduler [50] for all CNN models.

In addition, CNN models were optimized using negative Pearson correlation as a loss function. The learning rate for PhysFormer [77] was set to 1×10^{-4} and it was optimized using a dynamic loss composed of several hyperparameters, a negative Pearson loss, a frequency cross-entropy loss and a label distribution loss as used by the authors and implemented in the rPPG-Toolbox [38]. Before computing HR for performance evaluation, both ground truth and estimated BVP signals were filtered using a bandpass filter (low cutoff = 0.60 Hz, high cutoff = 3.30 Hz) to accommodate HR ranges of 36 to 198 BPM. HR was then computed using the FFT-peaks-based approach as implemented in rPPG-Toolbox [38].

A.3 Ablation Studies for FactorizePhys

We conduct ablation studies to evaluate optimal architectural choices and hyperparameters for the proposed FactorizePhys and FSAM. In table 3, we compare base FactorizePhys without FSAM and with FSAM and observe consistent performance gains with FSAM. Evaluation with and without residual connection indicates performance gains when residual connection around FSAM is implemented.

Table 3: Ablation study to assess residual connection to FSAM Module, and to compare the models trained with FSAM, for their inferences without FSAM

Training Dataset	**Testing Dataset**	**Training**	**Inference**	**MAE (HR) ↓**	**RMSE (HR) ↓**	**MAPE (HR) ↓**	**Corr (HR) ↑**	**SNR (BVP) ↑**	**MACC (BVP) ↑**
UBFC-rPPG	PURE	Base	Base	1.37 ± 1.02	7.97 ± 2.84	2.55 ± 2.07	0.94 ± 0.04	13.74 ± 0.81	0.77 ± 0.02
		Base + FSAM	Base + FSAM	0.71 ± 0.39	3.05 ± 1.02	1.20 ± 0.76	0.99 ± 0.02	13.78 ± 0.81	0.77 ± 0.02
			Base	0.71 ± 0.39	3.05 ± 1.02	1.20 ± 0.76	0.99 ± 0.02	13.78 ± 0.81	0.77 ± 0.02
		Base + FSAM + Res	Base + FSAM + Res	**0.48 ± 0.17**	**1.39 ± 0.35**	**0.72 ± 0.28**	**1.00 ± 0.01**	**14.16 ± 0.83**	**0.78 ± 0.02**
			Base	**0.48 ± 0.17**	**1.39 ± 0.35**	**0.72 ± 0.28**	**1.00 ± 0.01**	**14.16 ± 0.83**	**0.78 ± 0.02**
	iBVP	Base	Base	1.99 ± 0.42	4.82 ± 1.03	2.89 ± 0.69	0.87 ± 0.05	5.88 ± 0.57	0.54 ± 0.01
		Base + FSAM	Base + FSAM	1.90 ± 0.34	3.99 ± 0.76	2.66 ± 0.50	**0.91 ± 0.04**	5.82 ± 0.57	0.54 ± 0.01
			Base	1.85 ± 0.33	**3.89 ± 0.75**	2.59 ± 0.49	**0.91 ± 0.04**	5.80 ± 0.57	0.54 ± 0.01
		Base + FSAM + Res	Base + FSAM + Res	**1.73 ± 0.39**	4.38 ± 1.06	**2.40 ± 0.57**	0.90 ± 0.04	**6.61 ± 0.58**	**0.56 ± 0.01**
			Base	1.74 ± 0.39	4.39 ± 1.06	2.42 ± 0.57	0.90 ± 0.04	6.59 ± 0.57	**0.56 ± 0.01**

Retention of performance gains despite FSAM being skipped during inference, for FactorizePhys trained with FSAM offers insight into the mechanics of how FSAM functions. This can be interpreted as follows: Optimization of a network having FSAM implemented as an attention mechanism influences the network to increase the saliency of the most relevant features, so that a factorized approximation of embeddings retains these features, while discarding the less important features. Due to the increased saliency of relevant features and the presence of residual connection, FSAM can be skipped during inference, significantly reducing computational overhead.

Table 4: Performance Evaluation of Models on PURE Dataset [55], Trained with UBFC-rPPG Dataset [2], using Different Ranks and Optimization Steps for Factorization

Optimization Steps	**Rank**	**MAE (HR) ↓**		**RMSE (HR) ↓**		**MAPE (HR)↓**		**Corr (HR) ↑**		**SNR (dB, BVP) ↑**		**MACC (BVP) ↑**	
for Matrix Factorization		**Mean**	SE	**Mean**	SE	**Mean**	SE	**Mean**	SE	**Mean**	SE	**Mean**	SE
Base		1.37	1.02	7.97	2.84	2.55	2.07	0.94	0.04	13.74	0.81	0.77	0.02
4	1	0.48	0.17	1.39	0.35	0.72	0.28	1.00	0.01	14.16	0.83	0.78	0.02
	2	1.40	1.02	7.98	2.84	2.59	2.07	0.94	0.04	13.71	0.81	0.77	0.02
	4	2.25	1.30	10.23	3.09	4.36	2.66	0.91	0.05	13.50	0.82	0.77	0.02
	8	1.44	1.02	7.98	2.84	2.64	2.07	0.94	0.04	13.70	0.83	0.77	0.02
	16	2.20	1.30	10.22	3.09	4.26	2.66	0.91	0.05	13.55	0.82	0.77	0.02
6	1	0.80	0.39	3.11	1.03	1.33	0.77	0.99	0.02	13.60	0.81	0.77	0.02
	2	1.31	0.84	6.55	2.30	2.45	1.72	0.96	0.04	13.42	0.81	0.76	0.02
	4	1.53	0.90	7.10	2.32	2.91	1.86	0.96	0.04	13.54	0.82	0.77	0.02
	8	2.22	1.30	10.23	3.09	4.29	2.66	0.91	0.05	13.75	0.82	0.77	0.02
	16	1.43	1.02	7.98	2.84	2.65	2.07	0.94	0.04	13.62	0.81	0.77	0.02
8	1	0.73	0.39	3.06	1.02	1.24	0.77	0.99	0.02	13.67	0.81	0.77	0.02
	2	1.44	1.02	7.98	2.84	2.64	2.07	0.94	0.04	13.35	0.82	0.77	0.02
	4	0.78	0.39	3.10	1.03	1.30	0.77	0.99	0.02	13.77	0.80	0.77	0.02
	8	0.73	0.39	3.06	1.02	1.24	0.77	0.99	0.02	13.50	0.82	0.77	0.02
	16	0.73	0.39	3.06	1.02	1.24	0.77	0.99	0.02	13.55	0.83	0.77	0.02

In table 4, we present results to compare the performance obtained for different ranks L, as well as the optimization steps used to solve factorization. For all experiments, FactorizePhys is trained with the UBFC-rPPG dataset [2] and the performance is presented for the PURE dataset [55]. We can observe that the best performance was achieved for rank $L = 1$ for the different steps used to solve the factorization. For higher ranks, performance remains on par with that of the network without the FSAM, indicating that for the rPPG estimation task, the rank-1 factorization offers the optimal spatial-temporal attention. These results align with the expected single source of the underlying BVP signals in different facial regions.

A.4 Statistical Significance of the Main Results

We performed repeated experiments with 10 different random seed values between 1 and 1000 to compare the proposed FactorizePhys trained with FSAM with the best performing SOTA rPPG method. For the cross-dataset generalization results reported in table 2, EfficientPhys with SASN [37] was found to perform the best among the existing SOTA methods.

Table 5: Performance Evaluation of Models on PURE Dataset, Trained with UBFC-rPPG Dataset, using Different Random Seed Values

Model	**Random Seed Value**	**MAE (HR) ↓**		**RMSE (HR) ↓**		**MAPE (HR)↓**		**Corr (HR) ↑**		**SNR (dB, BVP) ↑**		**MACC (BVP) ↑**	
		Mean	**SE**	**Mean**	**SE**	**Mean**	**SE**	**Mean**	**SE**	**Mean**	**SE**	**Mean**	**SE**
EfficientPhys with SASN Attention Module	10	3.75	1.62	12.97	3.53	5.69	2.52	0.84	0.07	8.60	0.99	0.65	0.02
	38	4.46	1.74	14.09	3.59	7.18	2.88	0.81	0.08	8.69	1.01	0.66	0.02
	55	4.67	1.79	14.55	3.65	7.50	2.97	0.80	0.08	8.69	1.01	0.66	0.02
	100	4.71	1.79	14.52	3.65	7.63	2.97	0.80	0.08	8.77	1.00	0.66	0.02
	128	4.74	1.79	14.52	3.65	7.68	2.97	0.80	0.08	8.84	0.99	0.66	0.02
	138	4.36	1.79	14.41	3.65	7.01	2.97	0.80	0.08	8.81	0.99	0.66	0.02
	212	4.52	1.78	14.42	3.65	7.37	2.96	0.80	0.08	8.64	0.99	0.66	0.02
	308	4.70	1.79	14.52	3.65	7.61	2.97	0.80	0.08	8.84	1.03	0.66	0.02
	319	4.70	1.79	14.55	3.65	7.55	2.97	0.80	0.08	8.96	1.00	0.66	0.02
	900	4.63	1.79	14.51	3.65	7.48	2.97	0.80	0.08	8.65	0.99	0.66	0.02
	Average	4.52	1.77	14.31	3.63	7.27	2.92	0.81	0.08	8.75	1.00	0.66	0.02
Proposed FactorizePhys with FSAM Attention Module	10	1.38	0.98	7.64	2.71	2.52	1.98	0.95	0.04	13.40	0.82	0.75	0.02
	38	4.31	1.86	14.93	3.79	7.11	3.18	0.79	0.08	12.52	0.84	0.75	0.02
	55	2.17	1.30	10.22	3.09	4.22	2.66	0.91	0.05	13.71	0.83	0.77	0.02
	100	0.48	0.17	1.39	0.35	0.72	0.28	1.00	0.01	14.16	0.83	0.78	0.02
	128	0.78	0.39	3.08	1.03	1.31	0.77	0.99	0.02	13.23	0.81	0.76	0.02
	138	0.52	0.19	1.56	0.40	0.72	0.27	1.00	0.01	13.03	0.80	0.76	0.02
	212	2.15	1.22	9.63	2.88	4.19	2.50	0.92	0.05	13.58	0.81	0.77	0.02
	308	1.50	0.98	7.70	2.71	2.79	1.99	0.95	0.04	13.39	0.82	0.77	0.02
	319	1.38	0.84	6.61	2.30	2.60	1.73	0.96	0.04	13.54	0.81	0.77	0.02
	900	3.34	1.70	13.46	3.69	5.21	2.78	0.83	0.07	12.76	0.83	0.76	0.02
	Average	1.80	0.96	7.62	2.30	3.14	1.81	0.93	0.04	13.33	0.82	0.76	0.02
Paired T Test		0.0001		0.0014		0.0001		0.0004		0.0000		0.0000	

For each random seed value, we trained the proposed FactorizePhys with FSAM and EfficientPhys with SASN [37] on the UBFC-rPPG [2] dataset and evaluated them on the PURE dataset [55]. Paired T tests for each reported evaluation metrics suggest that the performance gains achieved with the proposed method are statistically significant compared against the best performing SOTA rPPG method, highlighting its effectiveness and thereby highlighting contributions of this work in the research field of end-to-end rPPG estimation from video frames.

A.5 Within Dataset Performance

In this work, we primarily focus on comparing rPPG methods for their cross-dataset generalization, which offers more critical evaluation and reliable estimates of how models perform on unseen or out-of-distribution data. Within-dataset performance signifies an representation ability of model to fit the data, derived from the same distribution, serving as an essential criteria. Therefore, for completeness, in table 6, we report within-dataset evaluation on iBVP [29], [55], and UBFC-rPPG [2] datasets, where we observe at-par performance of FactorizePhys as compared with the SOTA rPPG methods.

Table 6: Within Dataset Performance Evaluation

Model	**Attention Module**	**MAE (HR) ↓**	**RMSE (HR) ↓**	**MAPE (HR)↓**	**Corr (HR) ↑**	**SNR (dB, BVP) ↑**	**MACC (BVP) ↑**
Performance Evaluation on iBVP Dataset, Subject-wise Split: Training (0.0 - 0.7), Test (0.7 - 1.0)							
PhysNet	-	1.18 ± 0.29	**2.10 ± 0.51**	1.64 ± 0.42	**0.98 ± 0.03**	10.63 ± 1.05	**0.68 ± 0.02**
PhysFormer	TD-MHSA*	1.96 ± 0.63	4.22 ± 1.47	2.49 ± 0.72	0.91 ± 0.07	**10.72 ± 1.04**	0.66 ± 0.03
EfficientPhys	SASN	2.74 ± 0.96	6.28 ± 2.14	3.56 ± 1.13	0.81 ± 0.10	7.01 ± 1.03	0.58 ± 0.03
EfficientPhys	FSAM (Ours)	1.30 ± 0.33	2.34 ± 0.60	1.75 ± 0.46	**0.98 ± 0.04**	7.83 ± 0.96	0.59 ± 0.02
FactorizePhys (Ours)	FSAM (Ours)	**1.13 ± 0.36**	2.42 ± 0.77	**1.52 ± 0.50**	0.97 ± 0.04	9.75 ± 1.05	0.65 ± 0.02
Performance Evaluation on PURE Dataset, Subject-wise Split: Training (0.0 - 0.7), Test (0.7 - 1.0)							
PhysNet	-	0.59 ± 0.27	1.28 ± 0.46	0.92 ± 0.44	**1.00 ± 0.02**	**19.66 ± 1.18**	**0.90 ± 0.01**
PhysFormer	TD-MHSA*	0.68 ± 0.26	1.31 ± 0.46	1.08 ± 0.43	**1.00 ± 0.02**	19.05 ± 1.07	0.87 ± 0.01
EfficientPhys	SASN	**0.49 ± 0.26**	**1.21 ± 0.46**	**0.73 ± 0.42**	**1.00 ± 0.02**	15.25 ± 1.20	0.80 ± 0.02
EfficientPhys	FSAM (Ours)	0.59 ± 0.27	1.28 ± 0.46	0.92 ± 0.44	**1.00 ± 0.02**	15.42 ± 1.25	0.80 ± 0.02
FactorizePhys (Ours)	FSAM (Ours)	**0.49 ± 0.26**	**1.21 ± 0.46**	**0.73 ± 0.42**	**1.00 ± 0.02**	19.63 ± 1.40	0.86 ± 0.01
Performance Evaluation on UBFC-rPPG Dataset, Subject-wise Split: Training (0.0 - 0.7), Test (0.7 - 1.0)							
PhysNet	-	**1.62 ± 0.73**	**3.08 ± 1.16**	**1.46 ± 0.68**	**0.98 ± 0.06**	5.21 ± 1.97	0.90 ± 0.01
PhysFormer	TD-MHSA*	1.76 ± 0.79	3.36 ± 1.30	1.60 ± 0.74	0.96 ± 0.08	6.10 ± 1.86	0.90 ± 0.01
EfficientPhys	SASN	2.30 ± 1.40	5.54 ± 2.53	2.28 ± 1.44	0.90 ± 0.13	6.75 ± 1.76	0.87 ± 0.01
EfficientPhys	FSAM (Ours)	2.91 ± 1.42	5.88 ± 2.52	2.79 ± 1.45	0.88 ± 0.14	**6.79 ± 1.82**	0.87 ± 0.01
FactorizePhys (Ours)	FSAM (Ours)	2.84 ± 1.42	5.87 ± 2.52	2.73 ± 1.46	0.88 ± 0.14	6.33 ± 2.00	**0.91 ± 0.01**

TD-MHSA*: Temporal Difference Multi-Head Self-Attention [77];

SASN: Self-Attention Shifted Network [37]; FSAM: Proposed Factorized Self-Attention Module

A.6 Scalability Assessment of FSAM

We further investigate FSAM for its scalability to higher spatial-temporal resolution. For this, we perform within-dataset evaluation on the UBFC-rPPG dataset [2], which is pre-processed with the regular input dimension of $160 \times 72 \times 72$ as well as with a higher spatial and temporal dimension of $240 \times 128 \times 128$. Repeatable experiments are conducted with 10 different random seeds between 1 and 1000 to compare the performance of FactorizePhys with FSAM for each spatial-temporal input dimension.

Comparable performance, as observed in table 7, for both spatial-temporal input dimensions, suggests that FSAM can be easily deployed for different spatial-temporal scales. It should also be noted that the higher spatial dimension of video frames (i.e., 128×128) does not produce improved performance, indicating that the spatial dimension of 72×72 is sufficient to extract rPPG signals with end-to-end methods.

Table 7: Scalability Assessment of FSAM for Higher Spatial and Temporal Dimensions

Input	Random	MAE (HR) ↓		RMSE (HR) ↓		MAPE (HR)↓		Corr (HR) ↑		SNR (dB, BVP) ↑		MACC (BVP) ↑	
Dimension	Seed Value	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE
160x72x72	10	2.84	1.43	5.87	2.52	2.73	1.45	0.88	0.14	6.49	2.03	0.90	0.01
	38	2.84	1.43	5.87	2.52	2.73	1.45	0.88	0.14	6.68	2.00	0.91	0.01
	55	2.97	1.41	5.89	2.52	2.84	1.44	0.88	0.14	6.52	1.98	0.91	0.01
	100	2.84	1.43	5.87	2.52	2.73	1.45	0.88	0.14	6.32	2.01	0.91	0.01
	128	2.97	1.41	5.89	2.52	2.84	1.44	0.88	0.14	6.42	1.99	0.91	0.01
	138	2.84	1.43	5.87	2.52	2.73	1.45	0.88	0.14	6.48	1.96	0.91	0.01
	212	2.91	1.42	5.88	2.52	2.79	1.45	0.88	0.14	6.40	1.99	0.91	0.01
	308	2.84	1.43	5.87	2.52	2.73	1.45	0.88	0.14	6.51	1.98	0.91	0.01
	319	2.91	1.42	5.88	2.52	2.79	1.45	0.88	0.14	6.44	2.03	0.91	0.01
	900	2.91	1.42	5.88	2.52	2.79	1.45	0.88	0.14	6.55	2.01	0.91	0.01
	Mean	2.89	1.42	5.88	2.52	2.77	1.45	0.88	0.14	6.48	2.00	0.91	0.01
240x128x128	10	3.04	1.92	7.56	3.64	3.22	2.18	0.83	0.17	6.68	1.93	0.90	0.01
	38	2.91	1.93	7.54	3.64	3.10	2.19	0.84	0.16	6.86	1.93	0.90	0.01
	55	2.97	1.92	7.54	3.64	3.16	2.18	0.84	0.16	6.63	1.91	0.91	0.01
	100	2.91	1.93	7.54	3.64	3.10	2.19	0.84	0.16	6.87	1.91	0.90	0.01
	128	3.11	1.91	7.56	3.64	3.28	2.17	0.84	0.17	6.71	1.87	0.90	0.01
	138	2.91	1.93	7.54	3.64	3.10	2.19	0.84	0.16	6.63	1.95	0.91	0.01
	212	3.04	1.92	7.56	3.64	3.22	2.18	0.83	0.17	6.81	1.93	0.90	0.01
	308	2.91	1.93	7.54	3.64	3.10	2.19	0.84	0.16	6.71	1.92	0.90	0.01
	319	3.04	1.92	7.56	3.64	3.22	2.18	0.83	0.17	6.83	1.93	0.91	0.01
	900	3.04	1.92	7.56	3.64	3.22	2.18	0.83	0.17	6.69	1.91	0.90	0.01
	Mean	2.99	1.92	7.55	3.64	3.17	2.18	0.84	0.17	6.74	1.92	0.90	0.01

A.7 Multimodal rPPG Extraction

As iBVP dataset offers synchronized RGB and thermal infrared video frames, we conducted a brief experiment using FactorizePhys with FSAM to investigate whether combining both modalities can result in performance gains for the estimation of rPPG. For this, we also individually trained FactorizePhys on RGB and thermal frames keeping the identical data split of 70%-30%. Results in table 8 suggest weaker presence of rPPG signal in thermal infrared frames, leading to poorer performance when FactorizePhys is trained only on thermal frames, while not showing significant performance gains when jointly trained with RGB and thermal frames.

Table 8: Performance Evaluation on iBVP Dataset, Subject-wise Split: Train (70%), Test (30%)

Modality of	MAE (HR) ↓		RMSE (HR) ↓		MAPE (HR)↓		Corr (HR) ↑		SNR (dB, BVP) ↑		MACC (BVP) ↑	
Input Frames	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE
T	6.40	0.97	8.58	2.11	8.66	1.26	0.84	0.09	-3.27	0.40	0.20	0.01
RGB	1.13	0.36	2.42	0.77	1.52	0.50	0.97	0.04	9.75	1.05	0.65	0.02
RGBT	1.10	0.36	2.42	0.77	1.49	0.50	0.97	0.04	9.65	1.04	0.64	0.02

A.8 Visual Overview of Cross-Dataset Generalization and Latency

Figure 5 offers a quick visual summary of the cross-dataset generalization performance on different evaluation metrics, their respective standard error, and latency for the proposed and existing SOTA methods. The performance reported on Y-axis of each plot is cumulative cross-dataset performance

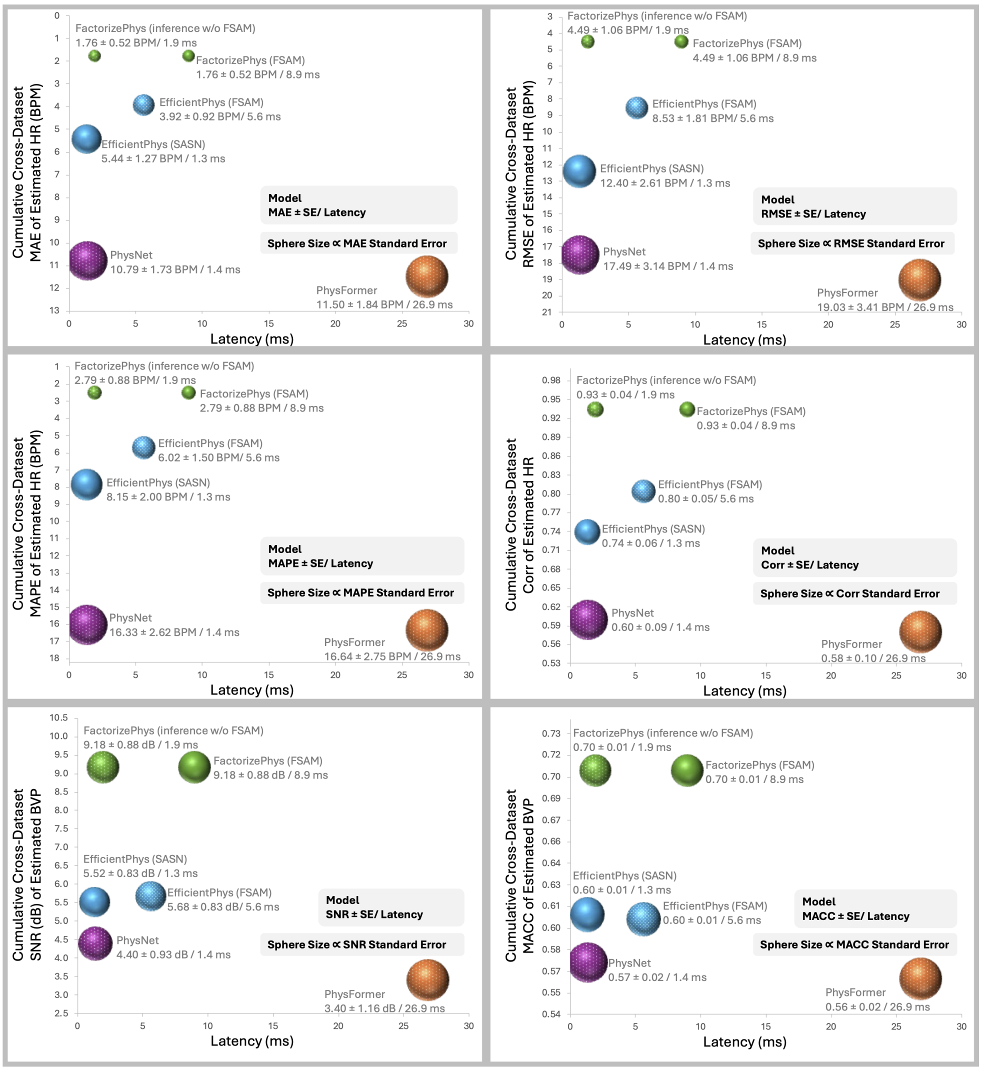

Figure 5: Cross-dataset performance comparison between SOTA and the proposed method reported with cumulative evaluation metrics, their standard error (SE) and latency

for respective models, averaged over different training and testing datasets. The proposed method outperforms existing state-of-the-art methods in all evaluation metrics by a significant margin, while achieving at-par latency.

A.9 Computational Cost and Latency

Table 9 compares model parameters, latency on GPU and CPU, and model size of the proposed FactorizePhys with FSAM with that of the existing SOTA rPPG methods. Considering the identical inference time performance of the base FactorizePhys, when trained using the proposed FSAM, the proposed method uses an order of magnitude fewer parameters and achieves a par latency on both CPU and GPU systems. Relatively higher latency compared to the EfficientPhys [37] model, despite the fewer model parameters, is due to the difference in the number of floating point operations (FLOPS). FactorizePhys, being a 3D-CNN architecture, requires more FLOPS to compute 3D features at each layer compared to the fewer FLOPS for EfficientPhys [37] which implements the 2D-CNN architecture. It should be noted that the FLOPS are also dependent on the input dimension, which is kept consistent for all the models. For resource critical deployment, FLOPS can be significantly reduced by decreasing the spatial dimension of input from 72 × 72 to 8 × 8 as found optimal for RTrPPG [3] or to 9 × 9 as used in the small branch of the Bigsmall model [44] for rPPG estimation.

Table 9: Comparison of FactorizePhys based on Model Parameters, Latency and Model Size

Model	**Model Parameters**	**Inference Time on CPU (ms)†**	**Inference Time on GPU (ms)‡**	**Model Size (MB)**
PhysFormer	7380871	450.47	26.86	29.80
PhysNet	768583	272.89	1.36	3.10
EfficientPhys with SASN	2163081	371.08	1.31	8.70
EfficientPhys with FSAM (ours)	140655	82.19	5.62	0.57
FactorizePhys Base (ours)	51840	96.80	1.94	0.22
FactorizePhys with FSAM (ours)	52168	95.75	8.97	0.22

†CPU Specs: Intel® Core™ i7-10870H CPU @ 2.20GHz × 16 GB RAM.

‡GPU Specs: NVIDIA GeForce RTX 3070 Laptop GPU (CUDA cores = 5120).

A.10 Visualization of Learned Attention

In fig. 6, we present additional samples of learned spatial-temporal features. For FactorizePhys trained with FSAM, we can observe superior cosine similarity and more relevant spatial distribution specifically under challenging scenarios with occlusions such as arising from hairs, eye-glasses and beard.

A.11 Qualitative Comparison with Estimated rPPG Signals

Qualitative comparison of the estimated rPPG signals between the proposed method and the best performing SOTA method (i.e., EfficientPhys [37] is presented for different test datasets - iBVP [29] (fig. 7) , PURE [55] (fig. 8), and UBFC-rPPG [2] (fig. 9).

A.12 Safeguards

We intend to release our rPPG estimation code only for academic purposes, with Responsible AI license (RAIL). Research areas that will benefit directly from this work include human-computer interaction and contactless health tracking or vital signs monitoring. Although the methods presented in this work may potentially benefit certain clinical scenarios, thorough validation studies, with appropriate ethics approval, are required to critically assess performance in such settings.

In addition, in some recent work, rPPG methods have been indicated as effective in detecting deepfake videos. In this context, we would like to caution such a use, considering the main results presented for the models trained using the SCAMPS [42] dataset, consisting of synthesized avatars. We argue that the rPPG signal can be embedded in the synthesized (or deep-fake) videos, with a similar approach as used for generating the SCAMPS [42] dataset. In such scenarios, in spite of high accuracy in estimating rPPG signals, such methods can be fooled by the synthesized videos that embed BVP signals. Therefore, we highlight that it is necessary to use the rPPG signal estimation methods in this context with great caution.

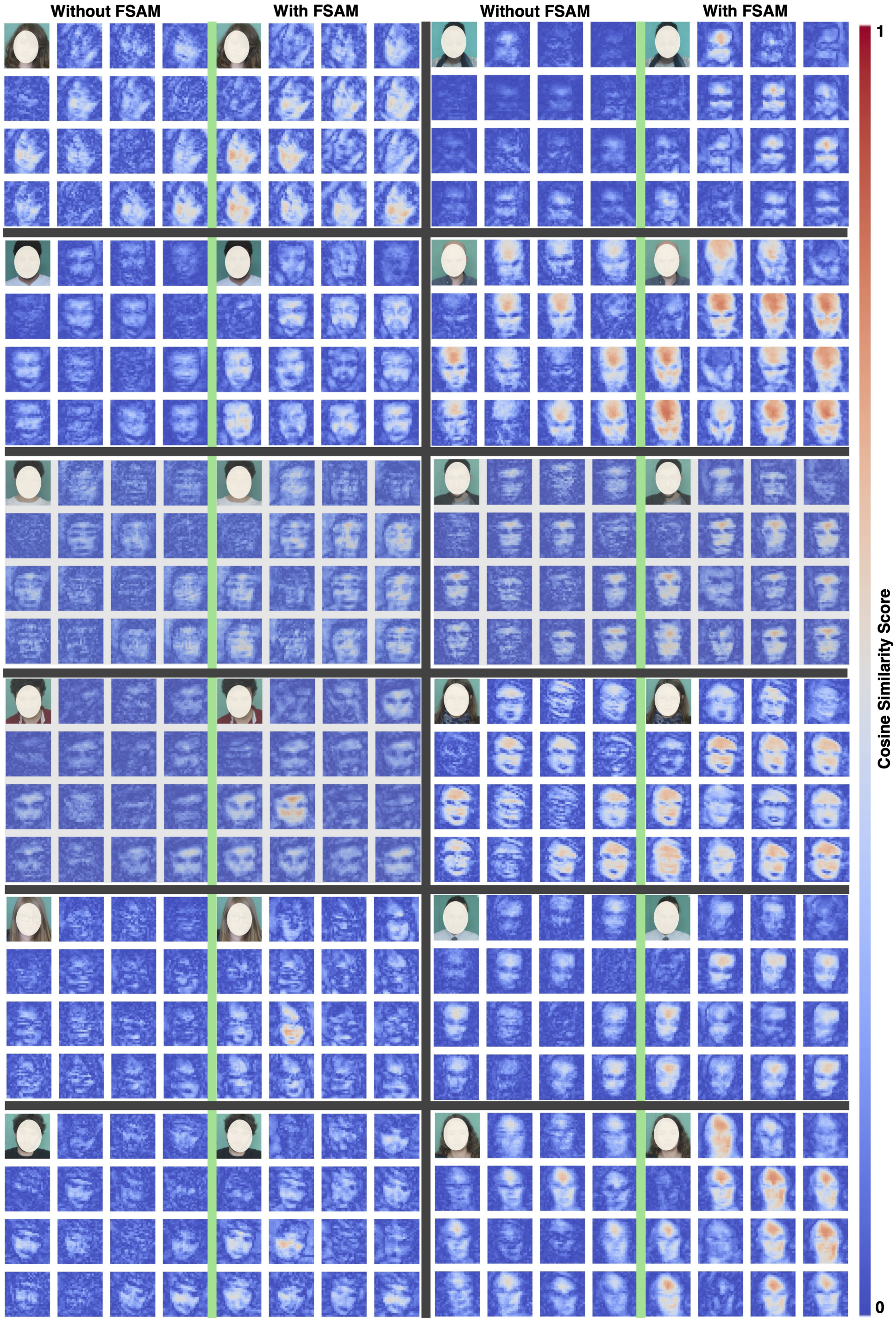

Figure 6: Visualization of Learned Spatial-Temporal Features

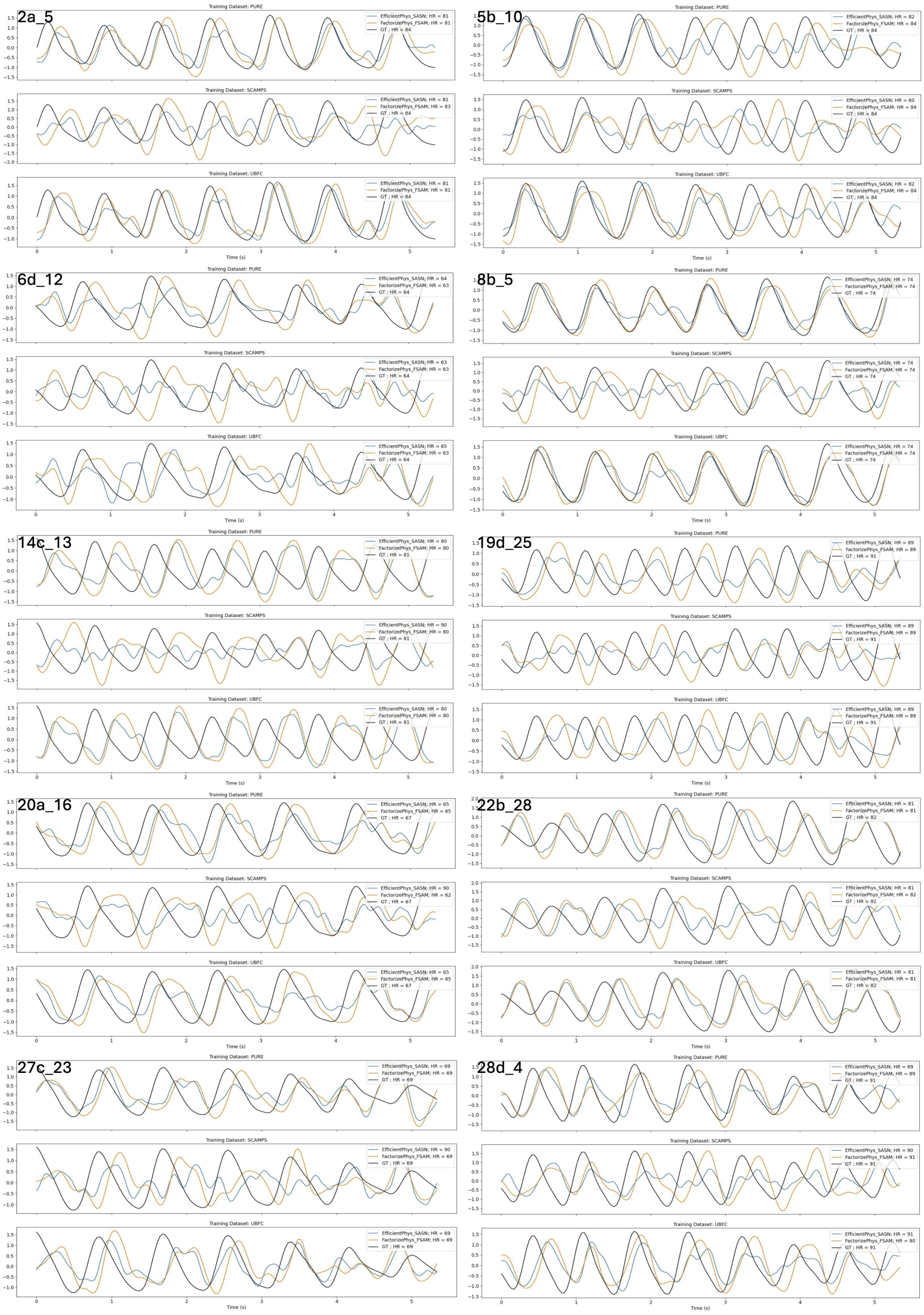

Figure 7: Comparison of Estimated rPPG Signals on iBVP Dataset for Models Trained with PURE, SCAMPS and UBFC-rPPG Datasets

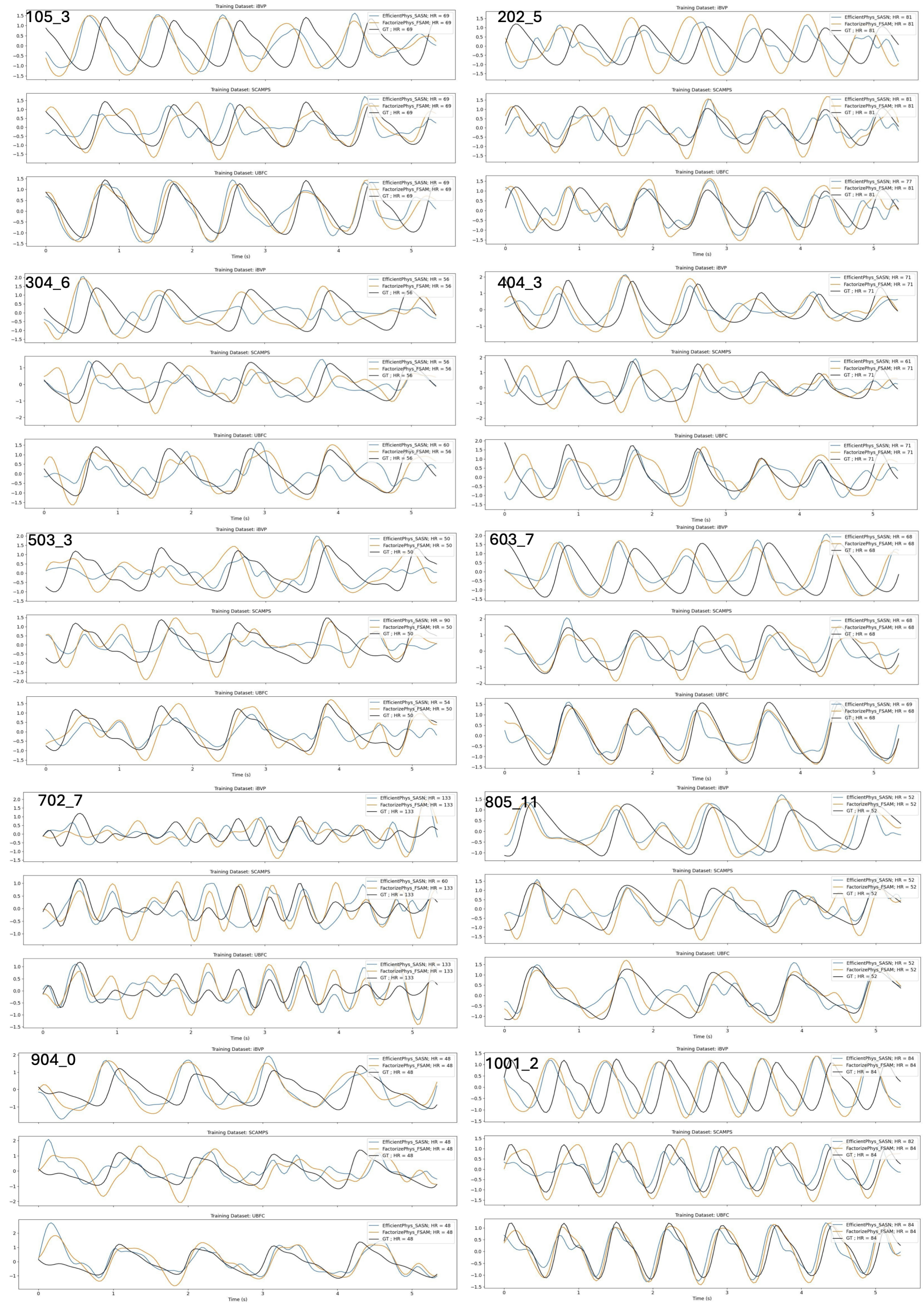

Figure 8: Comparison of Estimated rPPG Signals on PURE Dataset for Models Trained with iBVP, SCAMPS and UBFC-rPPG Datasets

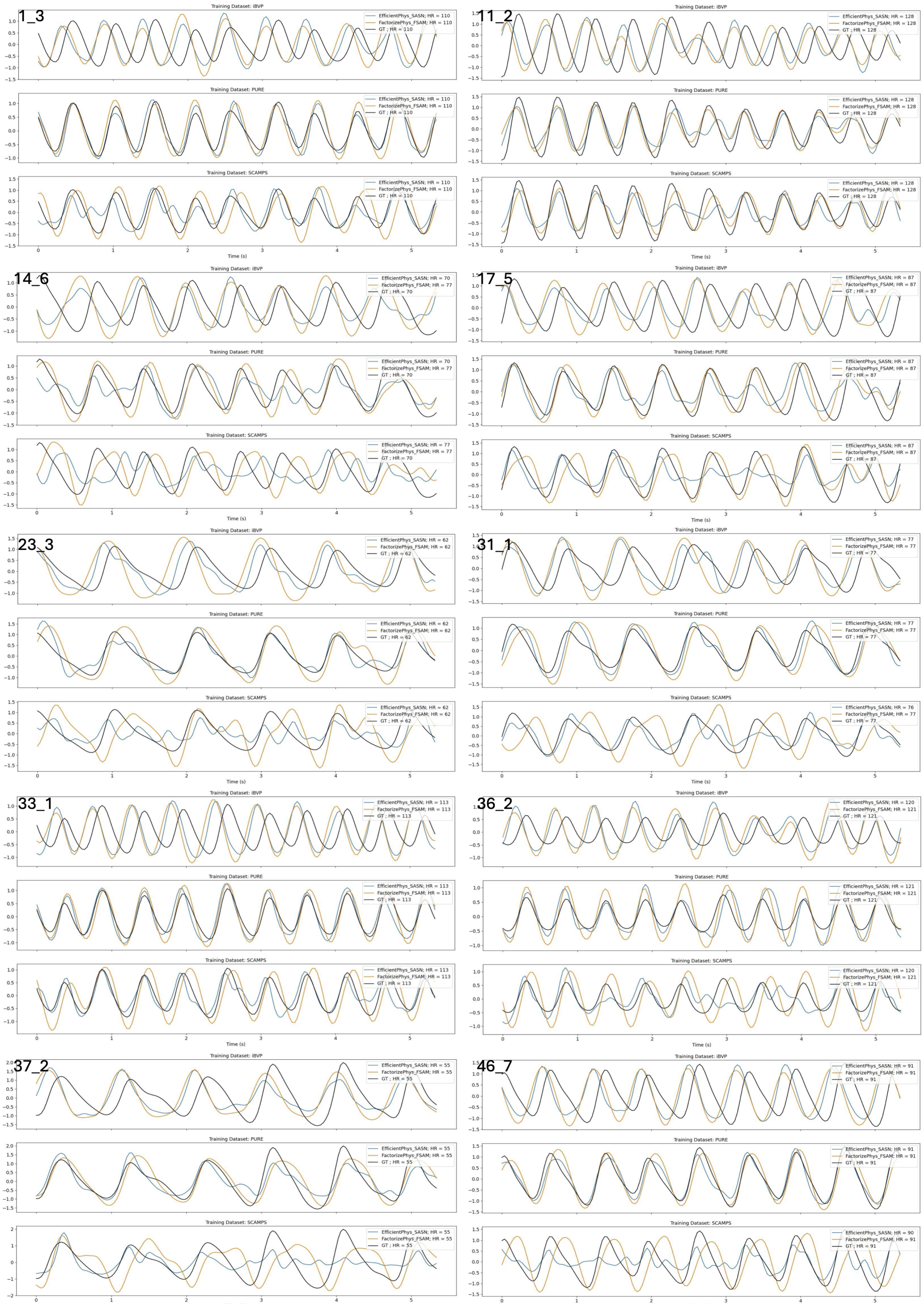

Figure 9: Comparison of Estimated rPPG Signals on UBFC-rPPG Dataset for Models Trained with iBVP, PURE and SCAMPS Datasets