Supplementary- ColFigPhotoAttnNet: Reliable Finger Photo Presentation Attack Detection Leveraging Window-Attention on Color Spaces

Anudeep Vurity[1], Emanuela Marasco[1], Raghavendra Ramachandra[2], Jongwoo Park[3]

[1]Center for Secure and Information Systems, George Mason University, U.S.A.
[2]Norwegian University of Science and Technology (NTNU), Gjøvik, Norway
[3]Stony Brook University, Stony Brook, NY, U.S.A

{avurity, emarasco}@gmu.edu, raghavendra.ramachandra@ntnu.no, jongwopark@cs.stonybrook.edu

1. Giga Multiply-Accumulate Operations

Giga Multiply-Accumulate Operations (GMAC), is a metric for computational complexity. It is used to measure the number of multiplication and addition operations a model makes during the phase of inference. This is an important metric to have, as it shows how computationally burdensome a model would be. GMACs provides a standard way to compare models based on their efficiency, especially in deep learning, where computations are intensive.

1.1. Explanation of GMACs for Different Models

The GMAC values for various models used in this paper are presented as follows:

GMAC values represents the computational complexity of each model. Table.1 shows all the Complexity of models and number of parameters used in this research. For instance, models like EfficientNet-B7 and SwinTransformer have high GMAC values: 5.17 and 4.38, respectively. Therefore, they have higher requirements for computational power, which, in turn, may give better performance but increase resource consumption. By contrast, models such as MobileNet-V3 Small (0.061 GMACs) and Mobile VIT-XXS (0.254 GMACs) have low GMACs—mostly used in scenarios where computational resources are limited.

Fig.1 shows the performance of all the applied models and the model complexity on each capture devices. *ColFigPhotoAttnNet* exhibits a balance between computational cost and performance. The figures depict the relationship between GMACs and BPCER (BonaFide Presentation Classification Error Rate) at a 5% APCER (Attack Presentation Classification Error Rate) for various models across iPhone, Google, Nokia and OPO capture devices.

The GMAC value for the *ColFigPhotoAttnNet* model is 1.79. Although *ColFigPhotoAttnNet* is more computationally demanding than some models, such as MobileNet-V3 Small, it was far less demanding compared with models like EfficientNet-B7 and Swin Transformer. This suggests that ColFigPhotoAttnNet is balanced between computational complexity and performance, which is suitable for applications that require fewer computational resources.

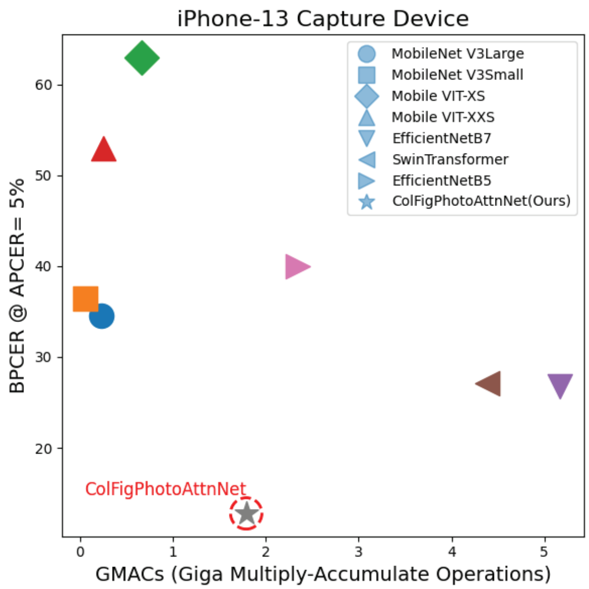
Baseline Models	**GMACs**	**Parameters (M)**
MobileNet V3Large	0.231	5.4
MobileNet V3Small	0.061	2.5
Mobile VIT-XS	0.665	2.3
Mobile VIT-XXS	0.254	1.3
EfficientNetB7	5.17	66.34
SwinTransformer	4.38	27.89
EfficientNetB5	2.35	30.38
ColFigPhotoAttnNet (Ours)	1.79	24.89

Table 1. Baseline models, GMACs, and Parameters

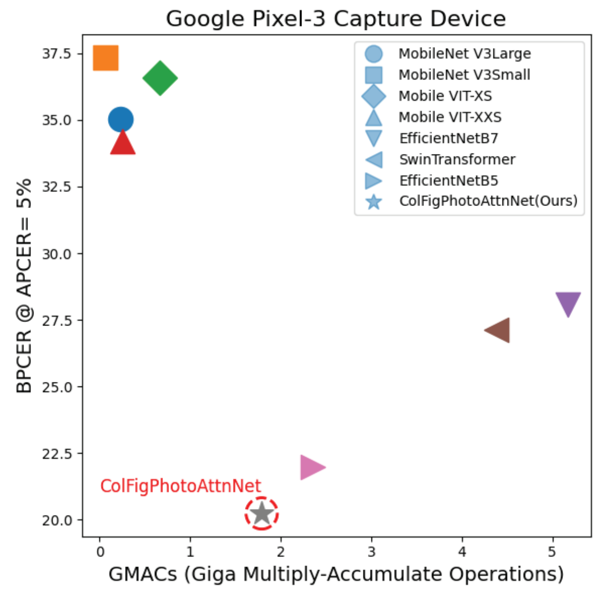
2. Additional Ablation Results

In this supplementary, we show the results of ablation studies on different datasets used. The Table.2 presents an ablation study using Google Pixel 3 capture device to evaluate the effectiveness of different model configurations in terms of BPCER at APCER thresholds: 5% and 10%. When only the RGB color space is used, the BPCER values are relatively high, with 22.91% at APCER 5% and 11.55% at APCER 10%. Adding the HSV color space alongside RGB reduces the BPCER to 15.64% and 7.45%, respectively. Including the YCbCr color space with RGB results in BPCER values of 17.58% and 8.64%. The inclusion of a bottleneck attention mechanism improves the model's performance. When using all three color spaces with bottleneck attention but without residual blocks yields a BPCER of 18.24% at APCER 5% and 5.91% at APCER 10%.

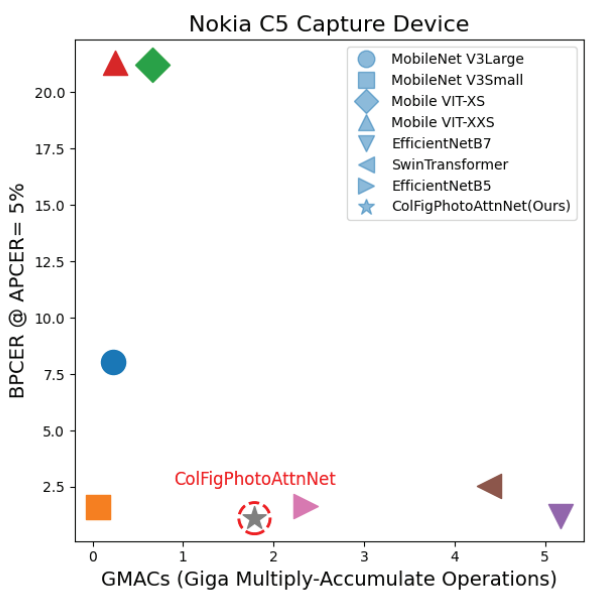
Similarly, in Table.3 presents an ablation study using Nokia capture device and Table 4 presents ablation study



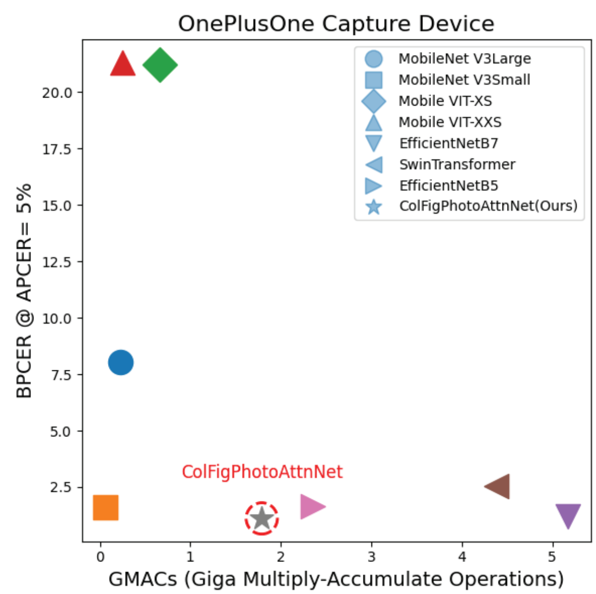
(a) GMACs vs Model Performance: iPhone13 capture device



(b) GMACs vs Model Performance: Google Pixel-3 capture device



(c) GMACs vs Model Performance: Nokia C5 capture device



(d) GMACs vs Model Performance: OnePlusOne capture device

Figure 1. The figure shows complexity vs Performance of the various models on all the different capture devices. ColFigPhotoAttnNet is highlighted with a red circle. a,b,c,d clearly shows that *ColFigPhotoAttnNet* shows performance vs complexity trade-off.

using OPO capture device. For both devices, the use of multiple color spaces generally improves model performance. The inclusion of bottleneck attention and residual blocks further enhances the model's accuracy. When tested on Nokia device, the configuration with all three color spaces, bottleneck attention, and residual blocks achieves a BPCER of 0.00% at both APCER 5% and 10%. Similarly, on the OPO device, the same configuration results in BPCER values of 0.64% and 0.00%, respectively. Figures 2 and 3 illustrate the performance of various models on Database 1 when using OPO and Nokia capture devices, respectively. The results clearly demonstrate that *ColFigPhotoAttnNet* model outperforms the baseline models in both scenarios.

Applying dynamic quantization (DQ) is found to reduce the model's complexity, although it slightly decreases performance. For instance, the configuration with DQ on the Nokia device shows a slight increase in BPCER compared to without DQ (0.11% vs. 0.00% at APCER 5%). On the

Color Spaces			BottleNeck	Residual	DQ	BPCER	BPCER
RGB	HSV	YCbCr	Attention	Block		5%	10%
✓	x	x	✓	✓	x	22.91	11.55
✓	✓	x	✓	✓	x	15.64	7.45
✓	x	✓	✓	✓	x	17.58	8.64
✓	✓	✓	✓	x	x	28.95	13.38
✓	✓	✓	x	✓	x	31.48	20.64
✓	✓	✓	✓	✓	x	18.24	5.91
✓	✓	✓	✓	✓	✓	20.24	6.73

Table 2. Ablation study using Database 2, This table shows BPCER @APCER 5% and APCER 10%.

Color Spaces			BottleNeck	Residual	DQ	BPCER	BPCER
RGB	HSV	YCbCr	Attention	Block		5%	10%
✓	x	x	✓	✓	x	1.24	0.25
✓	✓	x	✓	✓	x	1.05	0.14
✓	x	✓	✓	✓	x	0.64	0.51
✓	✓	✓	✓	x	x	3.74	1.41
✓	✓	✓	x	✓	x	1.21	0.21
✓	✓	✓	✓	✓	x	0.00	0.00
✓	✓	✓	✓	✓	✓	0.11	0.19

Table 3. Ablation study using Nokia Capture Device, This table shows BPCER @APCER 5% and APCER 10%.

Color Spaces			BottleNeck	Residual	DQ	BPCER	BPCER
RGB	HSV	YCbCr	Attention	Block		5%	10%
✓	x	x	✓	✓	x	6.91	2.85
✓	✓	x	✓	✓	x	3.53	1.45
✓	x	✓	✓	✓	x	4.85	1.24
✓	✓	✓	✓	x	x	1.08	0.54
✓	✓	✓	x	✓	x	1.81	1.62
✓	✓	✓	✓	✓	x	0.64	0.00
✓	✓	✓	✓	✓	✓	1.11	0.21

Table 4. Ablation study using OPO Capture Device, This table shows BPCER @APCER 5% and APCER 10%.

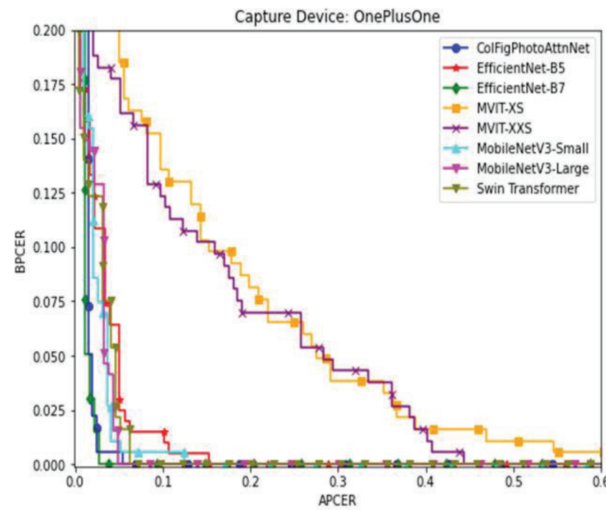


Figure 2. OPO capture device

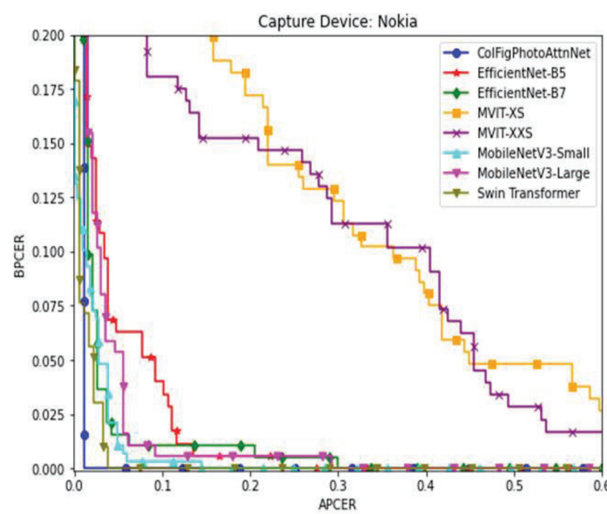


Figure 3. Nokia capture device

OPO device, a similar trend is observed, with BPCER values of 1.11% at APCER 5% and 0.21% at APCER 10% when DQ is applied.

R2-Q2: Secondly, I would suggest that you present your results in a more balanced manner. While it is impressive that ColFigPhotoAttnNet outperforms other models on some capture devices, the lack of convincing evidence of its generalization ability across different datasets and scenarios is concerning. I would recommend including additional experiments or using existing datasets to demonstrate the model's robustness and ability to generalize.

Yes, Efficient-b7, Efficient-b5 has shown better perofrmance on OPO dataset on intra class study. Therefore, as reviewer suggested We added additional intra class results Leave One Out method.

The results and observations are given below:

- The inclusion of HSV and YCbCr alongside RGB consistently shows better results compared to using RGB alone. When using multiple color spaces improves model performance. Configurations that include RGB, HSV, and YCbCr tend to yield lower BPCER values at both APCER thresholds (5% and 10%).
- The inclusion of bottleneck attention mechanisms significantly enhances model performance across all devices. Residual blocks further improve performance

when used in conjunction with bottleneck attention. Models with both bottleneck attention and residual blocks achieve the lowest BPCER values in most cases. For instance, the configuration with all three color spaces, bottleneck attention, and residual blocks yields the best results on both the Nokia and OPO devices.

- Dynamic Quantization (DQ) decreases the model complexity but marginally increases the BPCER%. Although there is a minor trade-off in BPCER%, this reduction in model complexity can be very useful for real-world applications with limited computational power.

3. Additional inter-capture results

This t-SNE plots (Fig.4 5) illustrates the inter- and intra-device generalization capabilities of most competing models, focusing on both old and new datasets. In this analysis, we use one new capture device (Google) and one older dataset (OnePlusOne) to evaluate how well ColFigAttnNet generalizes across devices. Specifically, we assess the model's performance when training and testing occur on the same device (intra-scenario) and when testing is performed on a different device (inter-scenario). EfficientNet-b5 and EfficientNet-b7 show competitive performance alongside ColFigAttnNet, prompting us to further investigate how these models generalize in similar scenarios. The t-SNE plot highlights the challenge of separating the live (blue) and spoof (orange) classes, with models showing varying levels of separation across different devices and testing conditions.

3.1. t-SNE analysis: Google Train

- The Fig.4 shows clear separation between live and spoof samples in the Google Train and Google Test scenario.This indicates that the model performs well in the intra-device scenario, learning features that generalize well when both training and testing are on the same device (Google).

- The separation is good even in the inter-device scenario, where the model was trained on Google but tested on iPhone. While the points are more dispersed, the separation is still visible, indicating that the framework has better generalization.

- EfficientNet b5 and b7 shows a heavy overlap between live and spoof classes in the iPhone Test set, showing that EfficientNet-b5 has difficulty in generalizing from Google Train to the iPhone Test. This inter-device scenario proves challenging for the model.

- There is significant overlap between live and spoof samples in the inter scenerio for both efficientNet-b5 and b7, indicating that both struggles to generalize in inter scenario.

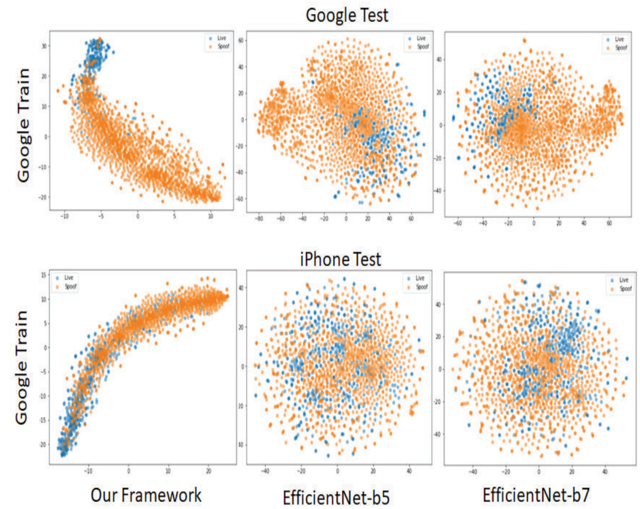


Figure 4. t-SNE plot illustrating the generalization performance in both inter- and intra-device scenarios for models trained on the Google dataset

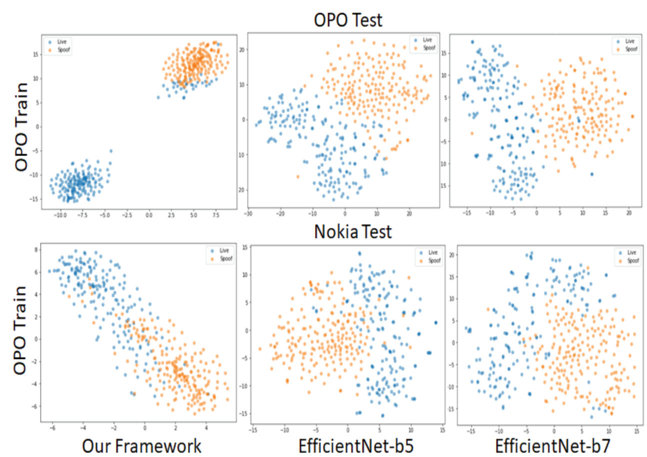


Figure 5. t-SNE plot illustrating the generalization performance in both inter- and intra-device scenarios for models trained on the OPO dataset.

3.2. t-SNE analysis: OPO Train

- Fig.5 shows that live and spoof classes are clearly well-defined in two different clusters. This indicates that the model performs exceptionally well when both the training and testing data come from the same OPO capture device.

- EfficientNet-b5 and EfficientNet-b5 shows weaker separations between live and spoof when compared to ColFigPhotoAttnNet in intra-scenario.

- In inter class scenario where the model was trained on OPO but tested on Nokia. We can see clusters are more dispersed compared to the intra-device scenario. Although the model still maintains a good separation between live and spoof classes, demonstrating its strong generalization ability across devices.

- In inter-class scenerio, although all the models have moderate saperation, EfficeintNet-B5 and B7 separation has noticeable overlap between the live and spoof classes compared to our framework. Both EfficientNet b5 and b7 struggles more in this inter-device scenario, indicating that there is some difficulty in generalization.

From the above analysis, we can see that our Framework is more robust in both intra-device and inter-device scenarios compared to other top models, showing better separation and generalization across devices. Additional investigation will be conducted in future work to provide a more comprehensive understanding of the model's effectiveness in diverse environments.