On the Performance of Composite 1D-to-2D Projections for Signal Quality Assessment

Guilherme Suzuki[1], **Pedro Garcia Freitas**[1]

[1] Department of Computer Science, University of Brasília, Brazil.

guilherme.chagas@aluno.unb.br, pedro.freitas@unb.br

Abstract. *Signal quality assessment is essential for health monitoring applications, as good signal quality is needed to reliably inform about the medical conditions of the patient. To achieve this, machine learning algorithms such as convolutional neural networks may be applied. However, the signal needs to be transformed into a 2D representation, which can be done using time series imaging techniques such as Gramian Angular Field (GAF), Markov Transition Field (MTF), and Recurrence Plot (RP), as well as by aggregating their results, which we refer to as Projection Mix. After preprocessing the dataset, Brno University of Technology Smartphone PPG (BUTPPG), into these images, various convolutional neural networks were trained and tested using such data, while also selecting hyperparameters through heuristic searching. The results indicate that our proposal performed better than the state-of-the-art methods.*

1. Introduction

Cardiovascular diseases such as coronary heart disease and stroke stand as a major global public health concern, claiming the lives of millions each year, as indicated by [Kaptoge et al. 2019]. In this context, the number of hypertensive patients will continue to increase worldwide, directly impacting any health system. In this scenario, the measurement of physiological parameters related to cardiac behavior, such as oximetry and heart rate, emerges as essential information that can be used for the control and prophylaxis of patients suffering from chronic diseases. While consistent monitoring of cardiovascular patterns is crucial for hypertensive individuals receiving home care or hospitalized patients, it necessitates precise and cost-effective equipment that prioritizes comfort. Currently, the gold standard method for conducting these measurements typically demands the expertise of a trained professional and provides an immediate assessment. However, it still exhibits various limitations attributed to its physical dimensions.

[Schmith et al. 2023] states that, considering public health concerns regarding cardiovascular diseases and practicality in clinical settings, numerous literature reports highlight automatic cuffless methods utilizing devices for measuring Photoplethysmogram (PPG) signals. With the advancement of Internet of Things (IoT) devices, wearable devices have become highly popular. They are now even smaller, smarter, more sensitive, scalable, and user-friendly. These devices can incorporate a wide range of sensing resources to continuously observe physiological functions. Among these sensing resources, PPG is a noteworthy sensor type since it can monitor various physiological parameters, such as heart rate, hypovolemia, blood oxygenation, respiration rate, and more. Moreover, PPG is worthy of attention because it is a simple optical method that consists of only two components placed on the skin. The first component is a light source used to

reflect light to the skin surface. The second component is a photodetector that collects the light reflection. As the PPG signal is composed of these optical components, it enables the continuous, non-invasive monitoring of physiological parameters. It is easy to implement, energy-efficient, and highly convenient for many clinical and commercial wearable devices.

Despite its undeniable advantages, PPG signals can be compromised by noises stemming from motion artifacts, signal acquisition, and other environmental sources. These noises have the potential to adversely affect signal quality, compromising the reliability of extracted health parameters. Such unreliable measurements could result in false alarms with potentially life-threatening consequences in healthcare applications. Consequently, Signal Quality Assessment (SQA) has emerged as an active research field in recent years. Several researchers have proposed techniques for evaluating PPG signals to analyze and optimize signal quality assessment performance.

In the literature, researchers investigated the PPG Signal Quality Index (SQI) by proposing assessment methods to discriminate 'reliable' and 'unreliable' parts of the signal. Among these methods, [Elgendi 2016] presents a comparison of eight evaluation metrics for signal quality based on perfusion, kurtosis, skewness, power, stationarity, zero crossing, entropy, and the matching of systolic wave detectors. It is a purely statistic-based approach. [Vadrevu and Manikandan 2019], [Reddy et al. 2020], and [Alam et al. 2021] described SQA using different signal features, also based on statistics, but considering additional constraints to enable the use of their methods in real-time applications on low-resource devices. Other studies presented machine learning approaches. For instance, [Li and Clifford 2012] employed a multi-layer perceptron to classify the quality status of PPG signals. [Pereira et al. 2019] investigated a set of machine learning approaches for quality assessment in 30-s segments of PPG, including k-nearest neighbors, decision trees, and support vector machines. Recently, deep learning methods based on 2D projections have emerged, converting the one-dimensional temporal-series raw PPG signal into bi-dimensional representations. [Naeini et al. 2023] employ one customized one-dimensional and three 2D Convolutional Neural Networks (CNN) to train models to assess PPG signals. In a similar fashion, [Freitas et al. 2023b, Freitas et al. 2023a] proposed a SQA method that converts PPG signals into two-dimensional representations, similar to 2D images. Subsequently, a vision transformer assesses the quality of these representations.

In this paper, we adopt a similar approach to the previously mentioned studies. Specifically, we extend the work of [Freitas et al. 2023a, Freitas et al. 2023b, Naeini et al. 2023] by incorporating all GAF, MTF, and RP projections within the same framework. Unlike previous works, our proposed approach combines these projection methods to create a composite, transforming them into a multi-channel (hyperspectral) image. In other words, we convert the temporal-series problem into a hyperspectral Computer Vision (CV) problem. To the best of our knowledge, no previous work has explored this composition of projections yet. Moreover, in contrast to the preceding papers, we consider the Computer Vision Classifier (CVC) as a hyperparameter of the overall framework. This means that, instead of using only Visual Transformer (ViT) as a classifier, we explore how different classifiers impact the performance of the overall framework, expanding the amount of possible CVC models that could be used for SQA.

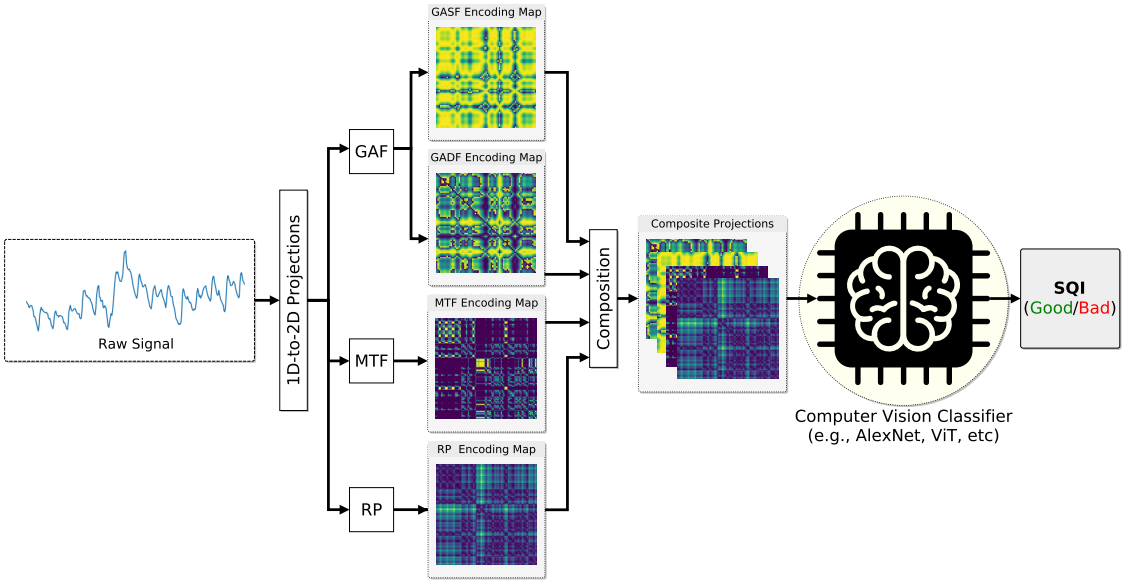


Figure 1. Block diagram of the proposed signal quality assessment pipeline. Different algorithms, namely GAF (with two variants, GASF and GADF), MTF, and RP project the one-dimensional raw signal onto two-dimensional image representations. Then, this hyperspectral image feeds a computer vision classifier. After training and evaluating the classifier, it can predict the signal quality indices (SQIs).

2. Proposed Method

Figure 1 depicts the summarized pipeline of the proposed method. This pipeline contains three main steps: (1) application of multiple 1D-to-2D signal projections; (2) hyperspectral image representation; and (3) image classification that employs a CVC to perform quality index estimation.

2.1. 1D-to-2D Projections

Mathematically, we can represent a PPG signal as a continuous function $X(t)$, where t is time. $X(t)$ represents the amplitude (intensity) of the PPG at any given point in time. In practice, sensors often sample the signal at discrete time points. Specifically, we may have a discrete representation $X[i]$, where i is the sample index. Thus, the PPG signal can be described as a time series $x = x_1, x_2, \cdots, x_n$ of n samples, where all values are in the interval $[-1, 1]$ and $x_i \in \mathbb{R}$. These one-dimensional temporal series can be projected onto a two-dimensional representation, such as GAF, MTF, and RP, offering advantages in visualization of temporal patterns, interpretability, applicability to computer vision techniques, and invariance to time warping.

Due to these advantages, in this paper, we combine three frameworks used to encode 1D PPG signals into 2D projections. The first type of projection, RP, represents a time series in a recurrence state in a phase space. It facilitates the representation of the phase-space trajectory in 2D, relying on its recurrence. The relationship between instants i and j is encoded in a 2D matrix, with elements assuming values of 0 or 1. The second type of projection, MTF, aims to generate a Markov matrix based on quantile bins, encoding transition probabilities within a quasi-Gramian Matrix. The third type of projection, GAF, represents time series data, such as a 1D signal, in a two-dimensional space by encoding

the pairwise angles between vectors in the original time series, visualized as an image.

2.1.1. Recurrence Plots

[Eckmann et al. 1987] introduced Recurrence Plots (RP) in 1987. In the original paper, the authors explored the use of recurrence as a tool for visualizing and analyzing the behavior of dynamical systems. Since then, RPs have become a valuable technique in the field of nonlinear time series analysis and have found applications in various scientific disciplines [Marwan 2008].

A RP is a projection that depicts the distances between trajectories extracted from the original time series. In a RP, the matrix elements correspond to the times at which a state of a dynamical system recurs. Specifically, columns and rows correspond to specific time pairs, capturing instances when the trajectory of the dynamical system traverses a region within the phase space that is approximately consistent.

Mathematically, for a signal x, the extracted trajectories are represented as $\vec{x}i = x_i, xi + \tau, \cdots, x_{i+\tau\cdot(m-1)}$, where $i \in 1, \cdots, n - (m-1)\cdot\tau$, n is the number of samples, m is the dimension of the trajectories, and τ is a time delay. The recurrence linking at instants i and j can be expressed by the following formula:

$$R_{ij} = \mathcal{H}(\varepsilon - \|\vec{x}_i - \vec{x}_j\|) = \begin{cases} 1 & \text{if } \|\vec{x}_i - \vec{x}_j\| \leq \varepsilon \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where $\mathcal{H}$ is the Heaviside step function, ε is a recurrence threshold, and $\|\cdot\|$ is a norm. Applying this expression creates a chart R that represents $\vec{x}_i = \vec{x}_j$ on the horizontal and vertical axes, respectively. It encodes recurrences through a binary depiction, where $Rij = 1$ signifies the presence of recurrence, and $Rij = 0$ indicates its absence.

2.1.2. Markov Transition Field

[Campanharo et al. 2011] introduced the MTF model to encode the statistics of dynamical transitions, preserving sequential Markov transition probabilities to retain temporal information. For the PPG signal x, the MTF identifies m quantile bins and assigns each x_i to the corresponding bin q_j. Consequently, the MTF constructs a weighted adjacency matrix W of size $m \times m$ by tracking transitions among quantile bins, akin to a first-order Markov chain along the time axis. The coefficients w_{ij} denote the frequency with which a point in quantile q_i succeeds a point in quantile q_j. Following normalization, where $\sum_j w_{ij} = 1$, the result is the Markov transition matrix. This matrix is designed to be insensitive to the signal distribution and temporal dependencies on time steps t_i. To retain crucial temporal dependencies and prevent excessive information loss in matrix W, MTF is given by

$$M = \begin{bmatrix} w_{ij|x_1\in q_i, x_1\in q_j} & \cdots & w_{ij|x_1\in q_i, x_n\in q_j} \\ w_{ij|x_2\in q_i, x_1\in q_j} & \cdots & w_{ij|x_2\in q_i, x_n\in q_j} \\ \vdots & \ddots & \vdots \\ w_{ij|x_n\in q_i, x_1\in q_j} & \cdots & w_{ij|x_n\in q_i, x_n\in q_j} \end{bmatrix} \tag{2}$$

The construction of the $m \times m$ Markov transition matrix W involves partitioning the magnitude into m quantile bins. The symbols q_i and q_j denote the quantile bins corresponding to the magnitude at instants i and j, respectively. This implies that the element M_{ij} in MTF matrix represents the probability transition from q_i to q_j. To clarify, the transition probability along the magnitude axis in the matrix W is translated into the MTF matrix by accounting for temporal positions.

2.1.3. Gramian Angular Field

[Wang and Oates 2015] introduced GAF as a technique for converting time-series data onto image data to obtain spatial correlation on time-series signals. This technique encompasses both the Gramian Angular Summation Field (GASF) and the Gramian Angular Difference Field (GADF). GASF represents the summation of the cosine values of the pairwise angles formed by vectors in the time series data. It provides a 2D projection where each element of the matrix corresponds to the cosine of the angle between the corresponding time points in the original series. GASF emphasizes the global patterns and similarities in the time series data. The resulting GASF projection is symmetric and captures overall trends. On the other hand, GADF represents the absolute differences between the sine values of the pairwise angles, focusing on capturing local variations and changes in the time series and highlighting details in the time series that might indicate transitions or fluctuations, making it sensitive to changes in the dynamics of the system. Through the conversion of one-dimensional time-series data into two-dimensional image data, the resultant projection image preserves the temporal correlations inherent in the original data. This enables the application of deep representation learning techniques to conduct deep feature learning and classification on the transformed image data.

Mathematically, The method initially rescales the original time-series data range to $[-1, 1]$ using

$$\tilde{x_i} = \frac{(x_i - \max(X)) + (x_i - \min(X))}{\max(X) - \min(X)}. \tag{3}$$

Then, the scaled data is converted into a polar coordinate system as expressed by

$$\begin{cases} \phi_i = \arccos(\tilde{x_i}), & -1 \leq \tilde{x_i} \leq 1 \\ r = \dfrac{t_i}{N}, & t_i \in \mathbb{N}, \end{cases} \tag{4}$$

where t_i is the timestamp, r stands for the polar coordinate radius, and N is a regularization factor to stretch over the polar coordinate system. Since the data scaling range is $[-1, 1]$, then the transformed angle range $\phi_i \in [0, \pi]$. It means that this conversion is a bijection, implying that it produces a unique result in the polar coordinate system and, therefore, it has a distinctive inverse mapping. Subsequently, this conversion executes a process akin to the inner product operation in Cartesian coordinates. The relationships between time points are established by computing the angles (both sum and difference)

between distinct sample points as follows:

$$
\begin{aligned}
GASF &= \tilde{x}^{\intercal} \cdot \tilde{x} - \left(\sqrt{\mathbb{1} - \tilde{x}^2}\right)^{\intercal} \cdot \sqrt{\mathbb{1} - \tilde{x}^2} \\
&= \{\cos(\phi_i + \phi_j)\}_{i,j} \\
&= \begin{bmatrix} \cos(\phi_1 + \phi_1) & \cdots & \cos(\phi_1 + \phi_n) \\ \cos(\phi_2 + \phi_1) & \cdots & \cos(\phi_2 + \phi_n) \\ \vdots & \ddots & \vdots \\ \cos(\phi_n + \phi_1) & \cdots & \cos(\phi_n + \phi_n) \end{bmatrix},
\end{aligned} \tag{5}
$$

and

$$
\begin{aligned}
GADF &= \left(\sqrt{\mathbb{1} - \tilde{x}^2}\right)^{\intercal} \cdot \tilde{x} - \tilde{x}^{\intercal} \cdot \sqrt{\mathbb{1} - \tilde{x}^2} \\
&= \{\sin(\phi_i - \phi_j)\}_{i,j} \\
&= \begin{bmatrix} \sin(\phi_1 - \phi_1) & \cdots & \sin(\phi_1 - \phi_n) \\ \sin(\phi_2 - \phi_1) & \cdots & \sin(\phi_2 - \phi_n) \\ \vdots & \ddots & \vdots \\ \sin(\phi_n - \phi_1) & \cdots & \sin(\phi_n - \phi_n) \end{bmatrix},
\end{aligned} \tag{6}
$$

where $\mathbb{1} = [1, 1, \cdots, 1]^{\intercal}$ the unit row vector with the same length of $\tilde{x}$. Following the conversion of the temporal-series signal into the polar coordinate system, two categories of GAF can be established utilizing the inner products $\langle x, y\rangle = x \cdot y - \sqrt{1 - x^2} \cdot \sqrt{1 - y^2}$ and $\langle x, y\rangle = \sqrt{1 - x^2} \cdot y - x \cdot \sqrt{1 - y^2}$. These inner products are quasi-Gramian matrices due to the non-linearity of the defined functions $\langle x, y\rangle$ within the inner-product space.

2.2. Signal Quality Classification

Figure 2 depicts two examples of PPG signals and how these temporal-series signals relate to the bi-dimensional projections. In this figure, the first row contains the waveforms of an 'unreliable' and a 'reliable' signal. The second row depicts their corresponding 2D projections produced via RP, MTF, and GAF algorithms. From these pictures, it is possible to notice that both RP, MTF, and GAF projections produced from the 'unreliable' PPG signal present more irregularity and chaotic visual patterns (higher entropy). On the other hand, the 'reliable' signal generates regular and symmetric visual patterns. These RP, MTF, and GAF encoding maps illustrate how distinct the 'reliable' and 'unreliable' quality status of the PPG signals can be compared to their 2D projections.

The correlation among the encoding maps of RP, MTF, and GAF with the quality of PPG signals was noted by [Freitas et al. 2023a, Freitas et al. 2023b]. In these works, the authors investigated how each of these projections, used individually, performs in the task of signal quality classification when employed as input for a ViT classifier. Despite the promising results, the authors investigated the performance of each projection individually. However, the combination of these projections remained an open question. In this paper, we investigate how this combination affects the performance of different classifiers in discriminating the quality of 'reliable' or 'unreliable' signals. For this purpose, we computed the projections separately for each of the MTF, GAF, and RP algorithms. In the case of GAF, we considered both GASF and GADF variants, described, respectively, in Equations 5 and 6 of Section 2.1.3.

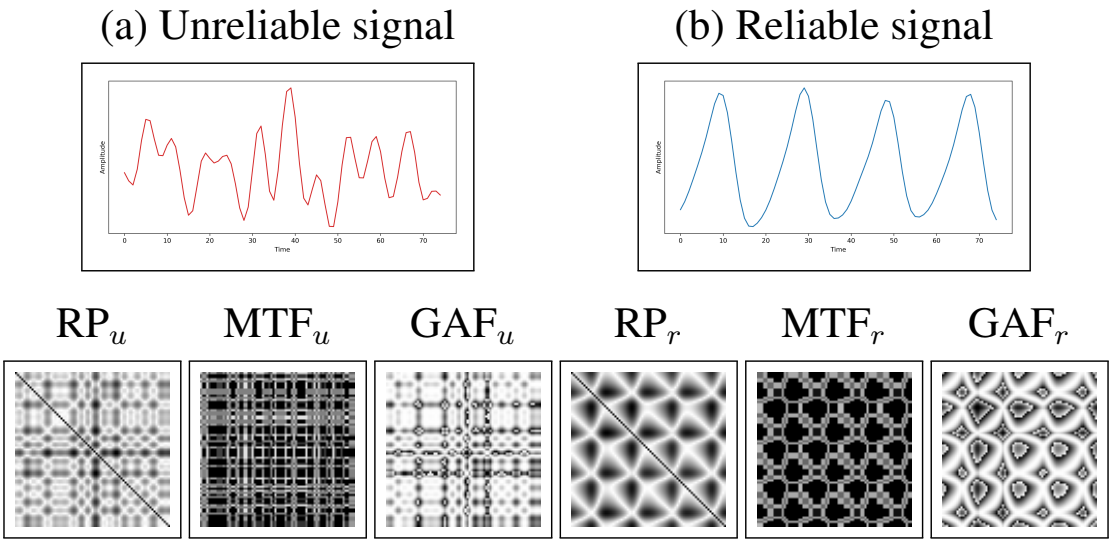


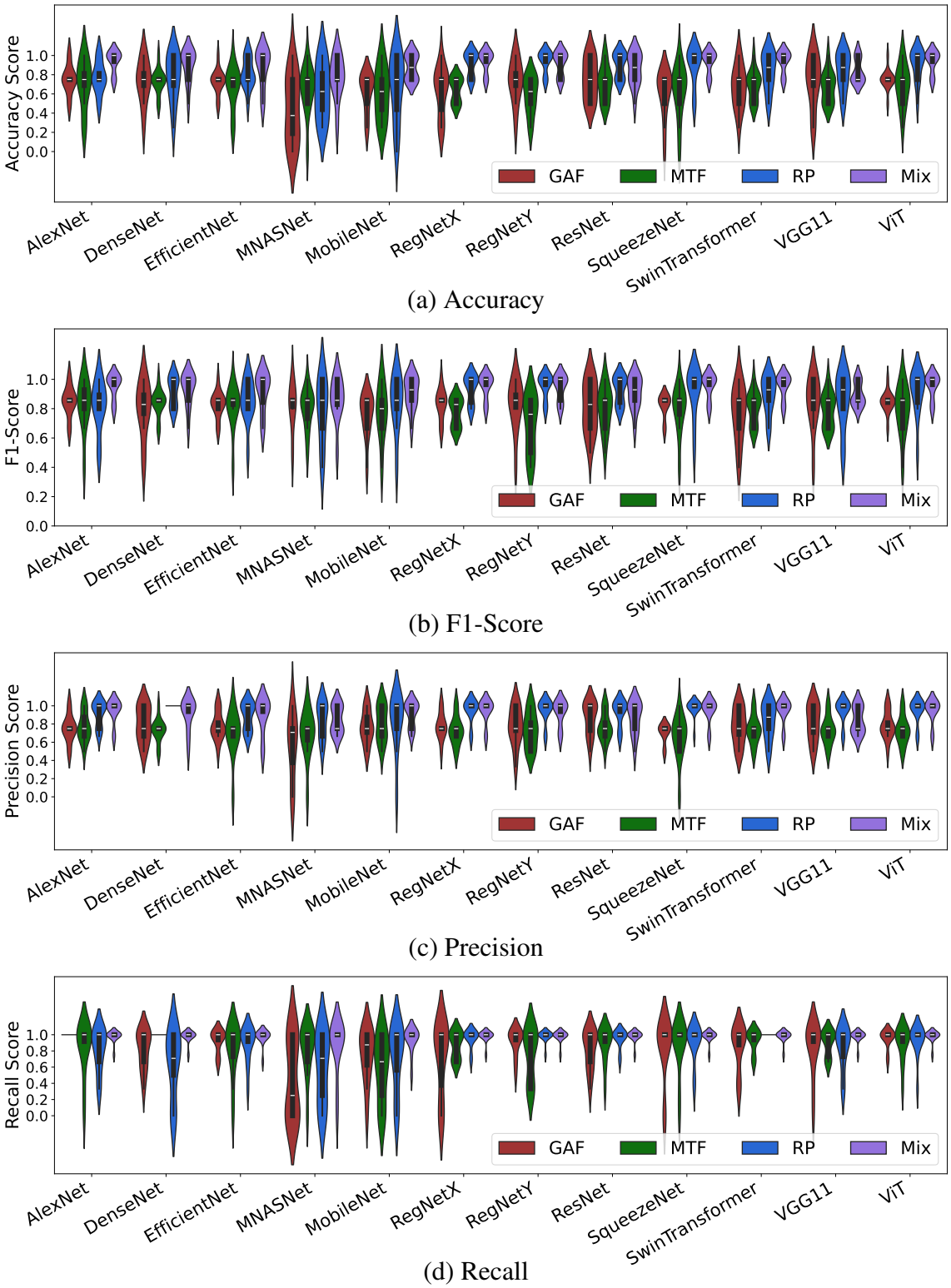
Figure 2. Example of signals and their corresponding 1D-to-2D encoding maps (projections). An 'unreliable' signal (a) generates asymmetrical encoding maps with less redundancy and higher visual variation, as we can observe from its RP_u, MTF_u, and GAF_u encoding maps. On the other hand, a 'reliable' signal (b) produces symmetric encoding maps with a more redundant pattern for all its RP_r, MTF_r, and GAF_r projections.

After generating the four independent projections (1 MTF, 1 RP, and 2 GAFs, one for GASF and one for GADF), the proposed method stacks those projections to compose a hyperspectral image (multi-channel). In other words, for a signal with n samples, the method transforms each of these four independent projections with dimensions $n \times n \times 1$ into a tensor of dimension $n \times n \times 4$. Then, we input these tensors into a CVC algorithm to predict the actual quality index (i.e., 'reliable' or not). In this work, we focus on deep-learning based CVCs, but other types could also be considered. To train the quality classifier, we map the input tensor to actual quality labels provided in the quality databases: $Q(PPG) = \mathfrak{C}(y, \mathcal{M})$, where $\mathfrak{C}(\cdot, \cdot)$ represents a CVCs algorithm, $\mathcal{M}$ is the trained model, and $Q(\cdot)$ is the quality score predicted using the model.

3. Experimental Results

We performed the tests using the BUTPPG database proposed by [Nemcova et al. 2021]. In this database, 48 PPG signals were recorded from the index fingers of 12 subjects. The recordings included three sessions while the subjects were seated and one session while they were walking [Nemcova et al. 2021]. This database can be accessed through the Physionet interface using the `wfdb` Python package. The PyTS library [Faouzi and Janati 2020] provided implementations for generating GAF, MTF, and RP representations. We used the PyTorch library [Paszke et al. 2019] for training and classification operations, focusing on classifying PPG signals. Additionally, we employed hyper-parameter optimization using the Optuna library [Akiba et al. 2019]. To balance the training dataset, we applied random oversampling using the imbalanced-learn library [Lemaître et al. 2017]. Furthermore, we utilized transfer learning by initializing the weights of neural networks pre-trained on the ImageNet dataset and fine-tuning them using the BUTPPG data.

The neural models chosen in our study were AlexNet, DenseNet, EfficientNet,



(a) Accuracy

(b) F1-Score

(c) Precision

(d) Recall

Figure 3. Violinplots illustrating the distribution of performance metrics for different models. Plots depict the variability scores across the model ensemble.

MNASNet, MobileNet, RegNetX, RegNetY, SqueezeNet, SwinTransformer, VGG11, and ViT. All these models were provided by TorchVision. Since the Projection Mix approach requires 4 channels in the input layer, we modified the models to meet this requirement.

Table 1. Performance Metrics for Neural Models with Different Projections. Considering each metric separately, the best results in terms of maximum absolute value are boldfaced, while the best results for each neural model contains an asterisk.

Neural model	Projection	Accuracy	F1 Score	Precision	Recall	Neural model	Projection	Accuracy	F1 Score	Precision	Recall
AlexNet	GAF	0.750	0.849	0.750	**1.000***	RegNetY	GAF	0.729	0.823	0.771	0.910
	MTF	0.708	0.827	0.773	0.833		MTF	0.583	0.690	0.729	0.757
	RP	0.771	0.819	0.910	0.812		RP	**0.938***	**0.955***	0.944*	0.979*
	Mix	**0.938***	**0.955***	0.944*	0.979		Mix	0.917	0.943	0.924	0.979*
DenseNet	GAF	0.729	0.795	0.799	0.819	ResNet	GAF	0.750	0.807	0.861	0.819
	MTF	0.729	0.837	0.729	**1.000***		MTF	0.708	0.799	0.771	0.889
	RP	0.771	0.917	**1.000***	0.646		RP	0.896*	0.926*	0.924*	0.951
	Mix	0.896*	0.932*	0.910	0.979		Mix	0.854	0.908	0.868	0.979*
EfficientNet	GAF	0.729	0.828	0.778	0.924	SqueezeNet	GAF	0.604	0.819	0.700	0.833
	MTF	0.667	0.806	0.694	0.812		MTF	0.625	0.794	0.646	0.861
	RP	0.812	0.856	0.882	0.889		RP	0.896	0.914	0.972*	0.903
	Mix	0.875*	0.916*	0.896*	0.972*		Mix	**0.938***	**0.955***	0.944	0.979*
MNASNet	GAF	0.438	0.812	0.552	0.458	SwinTransformer	GAF	0.667	0.765	0.771	0.847
	MTF	0.646	0.798	0.681	0.819		MTF	0.688	0.806	0.736	0.924
	RP	0.625	0.787	0.843	0.590		RP	0.833	0.897	0.833	**1.000***
	Mix	0.771*	0.866*	0.848*	0.861*		Mix	**0.938***	**0.955***	0.944*	0.979
MobileNet	GAF	0.625	0.758	0.765	0.743	VGG11	GAF	0.729	0.840	0.811	0.833
	MTF	0.604	0.777	0.806	0.583		MTF	0.667	0.790	0.729	0.896
	RP	0.667	0.838	0.818	0.715		RP	0.833*	0.855	0.944*	0.840
	Mix	0.875*	0.908*	0.910*	0.938*		Mix	0.833*	0.895*	0.840	0.979*
RegNetX	GAF	0.625	0.831	0.778	0.708	ViT	GAF	0.750	0.844	0.785	0.951
	MTF	0.646	0.773	0.729	0.868		MTF	0.688	0.790	0.729	0.889
	RP	0.917	0.938	0.944*	0.951		RP	0.896	0.913	0.944*	0.924
	Mix	**0.938***	**0.955***	0.944*	0.979*		Mix	**0.938***	**0.955***	0.944*	0.979*

Table 2. Memory usage of each model in terms of its parameters and buffers.

Neural Network	Memory Size (MB)	Neural Network	Memory Size (MB)	Neural Network	Memory Size (MB)	Neural Network	Memory Size (MB)
AlexNet	228	MNASNet	3	RegNetY	15	SwinTransformer	110
DenseNet	27	MobileNet	16	ResNet	44	VGG11	515
EfficientNet	16	RegNetX	20	SqueezeNet	2	ViT	349

For most of them, we added a convolution layer with a kernel size of 1 and 4 input channels so that its output dimensions match those of the original first layer. In the case of RegNet, we adjusted the number of input channels in the stem layer. For the standard projection methods, we replicated the projection into the 3 input channels without adding extra layers.

For the purpose of testing those models, we separated the dataset using a k-fold cross-validation training-test strategy. This strategy divides the dataset into k sets of equal size. The model is then trained using $k-1$ of the folds and validated in the remaining kth fold. Figure 3 and Table 1 present the results of this approach. Figure 3 contains violin plots where the correlation score of each fold for a given metric is a point of the distribution. In that sense, if the shape of the violin is larger at a certain location and its length is shorter, it indicates consistent model performance across different measurements. Such behavior is very recurrent in the Mix method, which achieves results where most fold scores concentrate in the median of all folds in all metrics. Furthermore, for a given model and metric, that median often achieves higher scores than the other projection

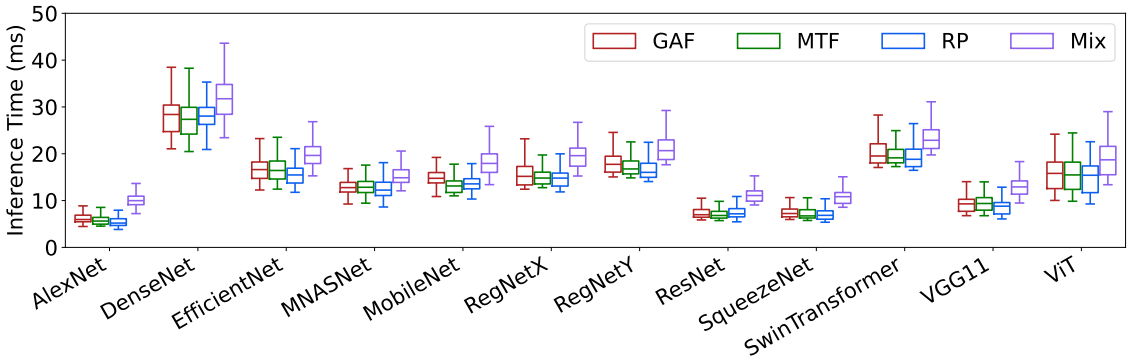


Figure 4. Boxplots where each point is the total inference time in milliseconds of each combination of neural model and projection method. The measured time includes the 1D-to-2D projection step.

methods, as seen in the accuracy score of AlexNet. These signs indicate that the proposed mix/composite method achieves higher performance. A visible exception occurs for the ResNet model, which achieves the best results with RP.

On the other hand, Table 1 reveals that, even though the PC mostly outperforms for the majority of the models, there are cases where other projections surpass it, such as for ResNet and RegNetY, where the Recurrence Plot obtains better results for most metrics. Additionally, the same table shows that the Mix method achieves the globally best scores for Accuracy and F1-Score, which contrasts with the fact that the other methods achieve the best Precision and Recall scores, but without achieving high scores in both metrics simultaneously.

Additionally, we measured the inference time and the memory consumption, as shown in Figure 4 and Table 2, respectively. Figure 4 was generated using 500 prediction/assessing time measurements. From this figure, it is noticeable that the speed of certain neural models stands out, such as the speed of AlexNet, ResNet, SqueezeNet, and VGG11. As for memory consumption, we see that some models consume a low amount of memory, such as MNASNet and SqueezeNet.

Considering all the information presented, a special case stands out: the combination of SqueezeNet and the Projection Composition (PC). In this case, Figure 3 shows that the PC is less scattered and has higher values than other approaches, showing a close competition with RP. However, Table 1 demonstrates that the PC has higher metric values than RP, except for Precision, where RP surpasses the PC, despite the PC having a value greater than 90%. Also, in Table 1, integrating SqueezeNet and PC gives the best average for the metrics Accuracy and F1 Score compared to all other combinations. Moreover, all metrics have values exceeding 90%. Finally, as seen before, this combination has high inference speed and low memory consumption.

The acceptance of these analyses needs to consider a series of limitations present in this procedure. Firstly, the dataset size is insufficient for training deep-learning models, which typically require large amounts of data. Secondly, we balanced the dataset using a simple technique, without exploring other alternatives specifically developed for time series data. Additionally, we did not explore the hyperparameters of the projections, such as the number of dimensions in the RP and the number of quantile bins in the MTF. Fur-

thermore, we only explored one of the multiple possible ways of constructing a composite projection from these individual projections.

4. Conclusion

This work presented a SQA method for PPG signals by projecting them into 2 dimensions using the RP, GAF, and MTF. In addition to these projection methods, we also proposed the use of a composition of them. Results indicate that the composition improves certain classification metrics and the generalization capabilities of the ensemble.

Acknowledgment

This work was supported by the Fundação de Apoio a Pesquisa do Distrito Federal (FAP-DF), by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), and by the University of Brasília (UnB). We thank Prof. Ricardo Lopes de Queiroz and Prof. Edson Mintsu Hung for letting us use their computing resources.

References

Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, page 2623–2631, New York, NY, USA. Association for Computing Machinery.

Alam, S., Gupta, R., and Sharma, K. D. (2021). On-board signal quality assessment guided compression of photoplethysmogram for personal health monitoring. *IEEE Transactions on Instrumentation and Measurement*, 70:1–9.

Campanharo, A. S., Sirer, M. I., Malmgren, R. D., Ramos, F. M., and Amaral, L. A. N. (2011). Duality between time series and networks. *PloS one*, 6(8):e23378.

Eckmann, J.-P., Kamphorst, S. O., and Ruelle, D. (1987). Recurrence plots of dynamical systems. *Europhysics Letters*, 4(9):973.

Elgendi, M. (2016). Optimal signal quality index for photoplethysmogram signals. *Bioengineering*, 3(4):21.

Faouzi, J. and Janati, H. (2020). pyts: A python package for time series classification. *Journal of Machine Learning Research*, 21(46):1–6.

Freitas, P., Lima, R., Lucafo, G., and Penatti, O. (2023a). Photoplethysmogram signal quality assessment via 1d-to-2d projections and vision transformers.

Freitas, P. G., De Lima, R. G., Lucafo, G. D., and Penatti, O. A. B. (2023b). Assessing the quality of photoplethysmograms via gramian angular fields and vision transformer. In *2023 31st European Signal Processing Conference (EUSIPCO)*, pages 1035–1039.

Kaptoge, S., Pennells, L., De Bacquer, D., Cooney, M. T., Kavousi, M., Stevens, G., Riley, L. M., Savin, S., Khan, T., Altay, S., et al. (2019). World health organization cardiovascular disease risk charts: revised models to estimate risk in 21 global regions. *The Lancet global health*, 7(10):e1332–e1345.

Lemaître, G., Nogueira, F., and Aridas, C. K. (2017). Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research*, 18(17):1–5.

Li, Q. and Clifford, G. D. (2012). Dynamic time warping and machine learning for signal quality assessment of pulsatile signals. *Physiological measurement*, 33(9):1491.

Marwan, N. (2008). A historical review of recurrence plots. *The European Physical Journal Special Topics*, 164(1):3–12.

Naeini, E. K., Sarhaddi, F., Azimi, I., Liljeberg, P., Dutt, N., and Rahmani, A. M. (2023). A deep learning–based ppg quality assessment approach for heart rate and heart rate variability. *ACM Transactions on Computing for Healthcare*, 4(4):1–22.

Nemcova, A., Vargova, E., Smisek, R., Marsanova, L., Smital, L., and Vitek, M. (2021). Brno university of technology smartphone ppg database (but ppg): Annotated dataset for ppg quality assessment and heart rate estimation. *BioMed Research International*.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. Curran Associates Inc., Red Hook, NY, USA.

Pereira, T., Gadhoumi, K., Ma, M., Liu, X., Xiao, R., Colorado, R. A., Keenan, K. J., Meisel, K., and Hu, X. (2019). A supervised approach to robust photoplethysmography quality assessment. *IEEE journal of biomedical and health informatics*, 24(3):649–657.

Reddy, G. N. K., Manikandan, M. S., and Murty, N. N. (2020). On-device integrated ppg quality assessment and sensor disconnection/saturation detection system for iot health monitoring. *IEEE Transactions on Instrumentation and Measurement*, 69(9):6351–6361.

Schmith, J., Kelsch, C., Cunha, B. C., Prade, L. R., Martins, E. A., Keller, A. L., and de Figueiredo, R. M. (2023). Photoplethysmography signal quality assessment using attractor reconstruction analysis. *Biomedical Signal Processing and Control*, 86:105142.

Vadrevu, S. and Manikandan, M. S. (2019). Real-time ppg signal quality assessment system for improving battery life and false alarms. *IEEE transactions on circuits and systems II: express briefs*, 66(11):1910–1914.

Wang, Z. and Oates, T. (2015). Imaging time-series to improve classification and imputation. In *Proceedings of the 24th International Conference on Artificial Intelligence*, pages 3939–3945.