Continuous Feature Networks: A Novel Method to Process Irregularly and Inconsistently Sampled Data With Position-Dependent Features

Birk Martin Magnussen
Opsolution GmbH
Ziegelstraße 17
34121 Kassel, Germany
e-mail: birk.magnussen@opsolution.de

Claudius Stern
FOM Hochschule für Oekonomie & Management
Hochschulzentrum Kassel
Garde-du-Corps-Straße 7
34117 Kassel, Germany
e-mail: Claudius.Stern@fom.de

Bernhard Sick
Intelligent Embedded Systems
Universität Kassel
Wilhelmshöher Allee 73
34121 Kassel, Germany
e-mail: bsick@uni-kassel.de

***Abstract*—Continuous Kernels have been a recent development in convolutional neural networks. Such kernels are used to process data sampled at different resolutions as well as irregularly and inconsistently sampled data. Convolutional neural networks have the property of translational invariance (e.g., features are detected regardless of their position in the measurement domain), which is unsuitable if the position of detected features is relevant for the prediction task. However, the capabilities of continuous kernels to process irregularly sampled data are still desired. This article introduces the continuous feature network, a novel method utilizing continuous kernels, for detecting global features at absolute positions in the data domain. Through a use case in processing multiple spatially resolved reflection spectroscopy data, which is sampled irregularly and inconsistently, we show that the proposed method is capable of processing such data directly without additional preprocessing or augmentation as is needed using comparable methods. In addition, we show that the proposed method is able to achieve a higher prediction accuracy than a comparable network on a dataset with position-dependent features. Furthermore, a higher robustness to missing data compared to a benchmark network using data interpolation is observed, which allows the network to adapt to sensors with a failure of individual light emitters or detectors without the need for retraining. The article shows how these capabilities stem from the continuous kernels used and how the number of available kernels to be trained affects the model. Finally, the article proposes a method to utilize the introduced method as a base for an interpretable model usable for explainable AI.**

***Index Terms*—machine learning; neural nets; continuous kernel; irregularly sampled data; reflection spectroscopy**

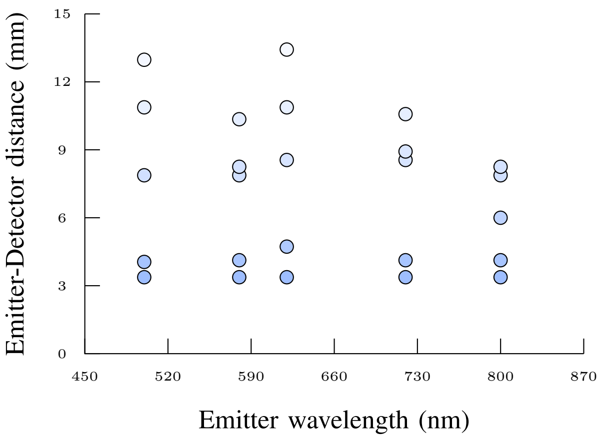
I. Introduction

This article is an extended version of the conference paper "Utilizing Continuous Kernels for Processing Irregularly and Inconsistently Sampled Data With Position-Dependent Features" [1] presented at the ICAS 2023 conference. Common machine learning methods assume that data is sampled consistently. That is, each instance of sampled data has the same shape and each data point always represents the same value. However, in real-world applications, data might often be sampled inconsistently due to factors like production inaccuracies of sensors measuring the data. In some cases, certain data points may be missing from the measurement as well. To facilitate such incomplete data, imputation of missing data points can be used to reconstruct the data [2]. Some methods also employ neural networks to reconstruct or correct data [3]. For certain types of network architectures, such as convolutional neural networks (CNN) [4], continuous kernels have been utilized to circumvent the need for special preprocessing and to handle irregularly and inconsistently sampled data directly instead [5] [6] [7]. Convolutional neural networks have the property of translational invariance, which is ill-suited to data expected to exhibit features at consistent absolute positions within the data sampling domain. Common examples of this type of data include spectral data, both optical and acoustic, where the relevant features may be intensity peaks at specific, consistent wavelengths rather than a wavelength-invariant feature of the intensity curve's shape.

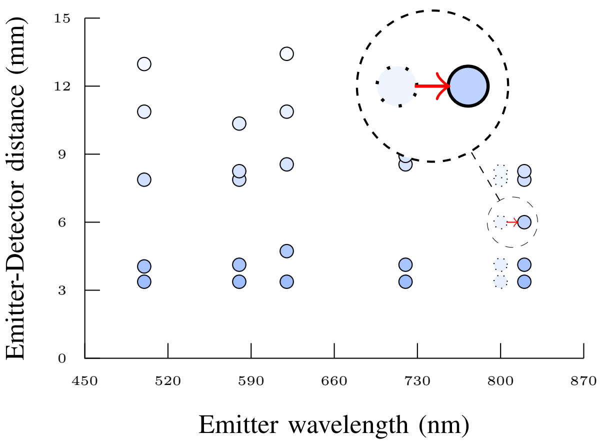
Certain types of spectrometry, such as multiple spatially resolved reflection spectroscopy (MSRRS, [8]), produce such data with position-dependant features, sampled both irregularly and inconsistently. An MSRRS-based sensor consists of several light emitters of different wavelengths, as well as several light detectors. These yield brightness values for all emitter-detector combinations, which have discrete emitter wavelengths and emitter-detector distance. An example of data from such a sensor can be seen in Figure 1. Figure 1a shows how such data is sampled irregularly due to the discrete nature of available emitters and detectors. Figure 1b shows one possible cause of inconsistently sampled data in MSRRS-based sensors. Due to production inaccuracies, it is possible that the wavelength of individual light emitters is slightly shifted, and the measured data points represent a different value in the measurement domain. Another possible cause of inconsistent data can be seen in Figure 1c, where data from one detector is unavailable.

To process irregularly and inconsistently sampled data with

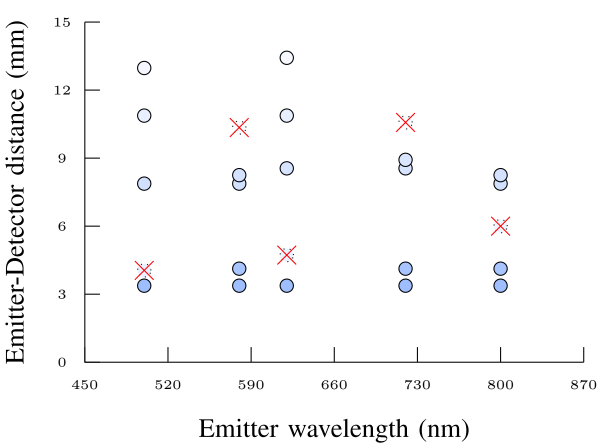
2023, © Copyright by authors, Published under agreement with IARIA - www.iaria.org



(a) Data from multiple spatially resolved reflection spectroscopy is only available at irregular intervals in the measurement domain.



(b) Production inaccuracies cause different emitter wavelengths for different devices.



(c) Individual detectors might fail, causing data to only be available inconsistently.

Fig. 1. Example of irregular and inconsistent data from a possible multiple spatially resolved reflection spectroscopy sensor arrangement. The data consists of relative brightness at discrete wavelengths and emitter-detector distances.

position-dependent features directly, a novel method is proposed in this paper. It has been shown that neural networks, such as sinusoidal representation networks (SIRENs) [9], can be used as functions parametrized through their learnable parameters, making them suitable for use as continuous kernels. These kernels are shown to be capable of modeling global, long-term dependencies [5]. By utilizing such continuous kernels outside of the context of convolutional neural networks, our method is capable of directly processing irregularly and inconsistently sampled data with position-dependent features.

The rest of the article is organized as follows. Section II details the current state-of-the-art regarding continuous kernels. Section III discusses the definition of a continuous kernel. Section IV introduces the new continuous feature networks utilizing continuous kernels. Section V introduces the experimental setup to evaluate the proposed method. The results are shown in Section VI, where the efficacy of the proposed method for processing spectroscopy data is discussed. Section VII discusses continuous feature networks as interpretable models for explainable AI. The conclusion closes the article.

II. Related Work

Previous work on continuous kernels focuses primarily on applications in CNNs to handle irregularly sampled data. In [5], continuous kernels are utilized to process various types of sequential data, including irregularly sampled sequences. The article contains an in-depth analysis of different types of continuous kernel parametrizations. [6] uses continuous kernels for convolutional neural networks to process non-grid bound data, such as representations of atoms in chemistry. In [7], continuous kernels are used to perform three-dimensional convolution on point clouds. However, as these articles all discuss the usage of continuous kernels for convolution in typical CNN architectures featuring certain degrees of translational invariance, the methods discussed are not well suited for use with data containing position-dependent features.

Using neural networks to represent a continuous function parametrized through the learnable parameters of the network, called implicit neural representation, has been previously analyzed by [10] [11] to model signed distance functions, which are required for shape representation of 3D geometry. In [9], a network architecture called SIREN is proposed as an implicit neural representation for generic data, including audio, images, and signed distance functions. Such implicit neural representations are useful to serve as continuous kernels [5].

III. Definition of Continuous Kernels

At its core, a continuous kernel is a function that assigns a weight to a data point at any given position [6]. Unlike in previous applications in CNNs, however, the position supplied to the kernel in our methods is an absolute position in the domain rather than a relative position to a convolution point.

To be able to represent continuous variants of typical weight kernels, the kernel function needs to be parametrizable with learnable parameters in such a way that the kernel function can ideally approximate any arbitrary function. It has been shown that multi-layer perceptrons (MPL) using sine nonlinearities, such as SIREN networks, can be used for such purposes.

2023, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Formalities

Small letters denote scalars, and small bold letters denote vectors. Capital letter variants of the former denote a set of the respective type. Subscripts on values indicate an index of the value within a containing set; superscripts indicate an index to the element of a vector.

Definition

Let $\mathbf{p}_i \in P \subset \mathbb{R}^n$ be the position of the value $d_i \in D \subset \mathbb{R}$ of the i-th data point of the set of data points D in an n-dimensional domain. A continuous kernel in the proposed architecture is now defined as a function

$$\psi : \mathbb{R}^n \mapsto \mathbb{R} \tag{1}$$

assigning a weight value to any position $\mathbf{p}_i$ in the domain. As shown in [5], such functions can be modeled and parameterized using implicit neural representations, such as MLPs using sine nonlinearities like SIREN [9]. In the proposed method, such an MLP serves as the function ψ. The MLP has n input neurons to input the absolute position $\mathbf{p}_i \in \mathbb{R}$ of a data point in the domain and one output neuron representing the assigned weight for the data point. The remaining model parameters of the kernel are the number and size of hidden layers in the MLP, which can be adjusted to the problem to be learned. The MLP serving as the weight function ψ of a continuous kernel is not trained separately but rather as part of the final network that the continuous kernel is used in.

IV. Continuous Feature Network

Figure 2 shows the general structure of the proposed architecture. Part I shows the set of input data points to the model, each representing a value at a specific position within the measurement domain. In the proposed method, the first layer of the architecture, called the continuous feature layer, contains multiple independent continuous kernels (see part II).

For each of the independent kernels, the input consisting of an arbitrary number of data points is weighed using the kernel. Additionally, the input data might be sampled unevenly. To compensate for an uneven distribution of samples, the local density of the sampled data points in the measurement domain is calculated (omitted in Figure 2). In the proposed method, kernel density estimation, where the kernel size is a learnable parameter, was used, but other methods for point density estimation can be used as well. Each data point is weighted by the inverse local density of data points at its position as proposed in [7]. The data points weighted by both the kernel and the inverse point density are shown in part III and are formally expressed in Equation (2).

For each kernel, the weighted data points are reduced to a single value as defined in Equations (3) and (4) and as shown in part IV. Here, a sum is used as the reduction operation, but other reductions, such as calculating the mean of the values, are also considerable. Combining the reduced value of each kernel into a vector results in an output feature vector of a fixed size depending on the number of independent kernels in the continuous feature layer.

Since the continuous feature layer reduces the input of arbitrary size to a latent vector of a fixed and predetermined size, the continuous feature layer can be followed with a typical neural network architecture, such as a multi-layer feed-forward network (see part V). The output of this MLP then serves as the output of the entire network as depicted in part VI. We call the proposed combination of a continuous feature layer followed by a multi-layer feed-forward network a continuous feature network. The continuous feature layer, as described, has three main model parameters: The number of kernels and the two parameters defining the shape of the kernels, being the number and the size of its hidden layers.

Formal Definition

Let the set Ψ be the set of multiple, independent continuous kernels ψ_k used in the continuous feature layer. In this set, each ψ_k represents one feature possibly present in the sampled data. Let $d_i \in D$ be the i-th data point in the input dataset D with the position $\mathbf{p}_i \in P$ in the measurement domain. Let $\rho(\mathbf{p}_i)$ denote the local density of sampled data at position $\mathbf{p}_i$ in the domain. Then we define the weighted data points for each kernel as

$$w_{i,k} := d_i \cdot \psi_k(\mathbf{p}_i) \cdot \frac{1}{\rho(\mathbf{p}_i)} \tag{2}$$

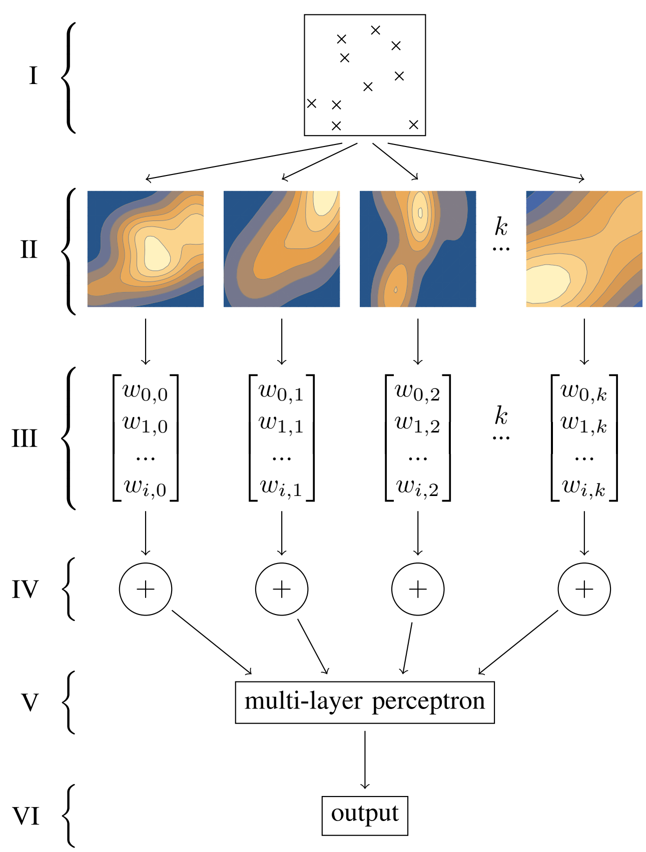


Fig. 2. Overview of a continuous feature network.

2023, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

The components of the resulting fixed-size latent feature vector $\mathbf{v}$ are defined as follows:

$$\mathbf{v}^k(D, P) := \sum_i (w_{i,k}) \tag{3}$$

$$= \sum_i \left(d_i \cdot \psi_k(\mathbf{p}_i) \cdot \frac{1}{\rho(\mathbf{p}_i)} \right) \tag{4}$$

As the fixed size feature vector $\mathbf{v}$ is a function of the data points and their position, the feature vector can be expressed as a function of the following type:

$$\mathbf{v} : \mathbb{R}^i, \mathbb{R}^{i \times n} \mapsto \mathbb{R}^k \tag{5}$$

$\mathbf{v}$ describes the feature vector with a fixed size k as a reduction of an input of arbitrary size i for the data and $i \times n$ for the data's position for any i. Since the size of the feature vector $\mathbf{v}$ is fixed and does not depend on the input size i, the feature vector can be used as the input to a classical neural network architecture, such as a multi-layer feed-forward network, for an arbitrary input size i without the need to retrain the network.

V. Experimental Setup

The method is tested with a dataset from an MSRRS-based sensor as described in [8]. The data was measured in vivo, alongside a reference measurement of the carotenoid concentration in the skin on a scale ranging from 0 to 12. The datasets for training and testing are entirely distinct, having measured a different group of test subjects using a different set of MSRRS-based sensors.

The measured spectroscopic data is well suited for the use of continuous kernels and continuous feature networks as proposed in Section IV. This is because the MSRRS-based optical data is yielded in the shape of a relative brightness given for certain discrete wavelengths and certain discrete distances between light-emitter-detector pairs. These discrete wavelengths and distances are neither sampled at regular intervals nor always at the same exact wavelengths. Due to production inaccuracies for the sensors, slight differences in the wavelength of the emitters exist. However, the peak wavelengths are known for each sensor's emitters and can thus be accurately supplied as the position data for the continuous feature layer. In addition, it is expected that the relevant data in this type of spectral data is encoded in the position of the features (here, the absorption wavelengths of the carotenoids) rather than the shape of features to be detected, rendering the proposed method suitable.

Evaluation Architectures

To evaluate the method, a continuous feature network with a continuous feature layer containing 64 continuous kernels is used. Each kernel is made up of a SIREN network, containing three hidden layers of 48 nodes with sine nonlinearities each. The continuous feature layer is followed by a hidden feed-forward layer with 64 nodes, followed by an output layer with one output for the predicted carotenoid concentration. This network has approximately 320k parameters.

For a comparison network, we use a multi-layer feed-forward neural network using a similar amount of parameters. This feed-forward network has one input node for each emitter-detector pair, followed by a hidden layer of 256 nodes, followed by another hidden layer of 128 nodes, followed by an output layer with one output for the predicted carotenoid concentration for a total of approximately 375k parameters.

In addition, a continuous feature layer without any subsequent network was examined. Instead, a simple learned weighted sum of the components of the latent feature vector was used as the model output.

A convolution-based model was also investigated, but it has proved unable to produce meaningful predictions of the carotenoid concentration in human skin and is thus omitted from further analysis in this article.

All networks were trained using the ADAM optimizer [12] and implemented using the LibTorch bindings of the PyTorch framework [13].

VI. Experimental Results

Figure 3 shows the accuracy of the proposed continuous feature network architecture compared to the accuracy of the comparison network. To show the ability of the continuous feature network to handle inconsistently sampled data, the prediction accuracy of the network was measured with the data of certain detectors withheld during inference. For each sample of data in the test set the detectors whose data was withheld were picked randomly, according to the number of detectors disabled.

For the continuous feature network and continuous feature layer, the missing data points were simply removed from the input vector. Due to the nature of the continuous feature network, it is capable of processing the shorter input vector without the need to retrain the model. For the multi-layer feed-forward network, the data was interpolated from the data of other detectors with a similar wavelength and emitter-detector

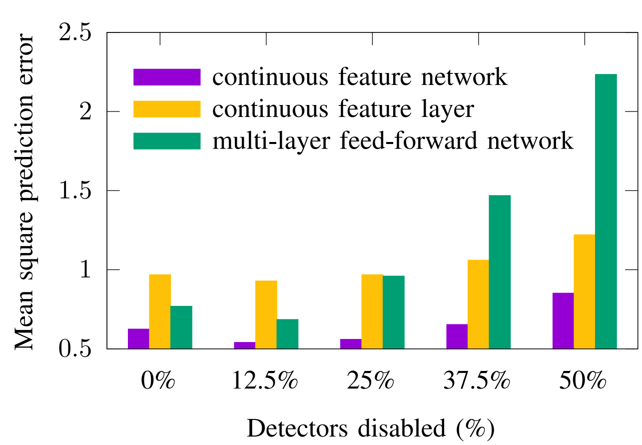


Fig. 3. The mean square prediction error (lower is better) of the continuous feature network, a continuous feature layer with no follow-up network, and the multi-layer feed-forward network with the data of a different number of detectors withheld.

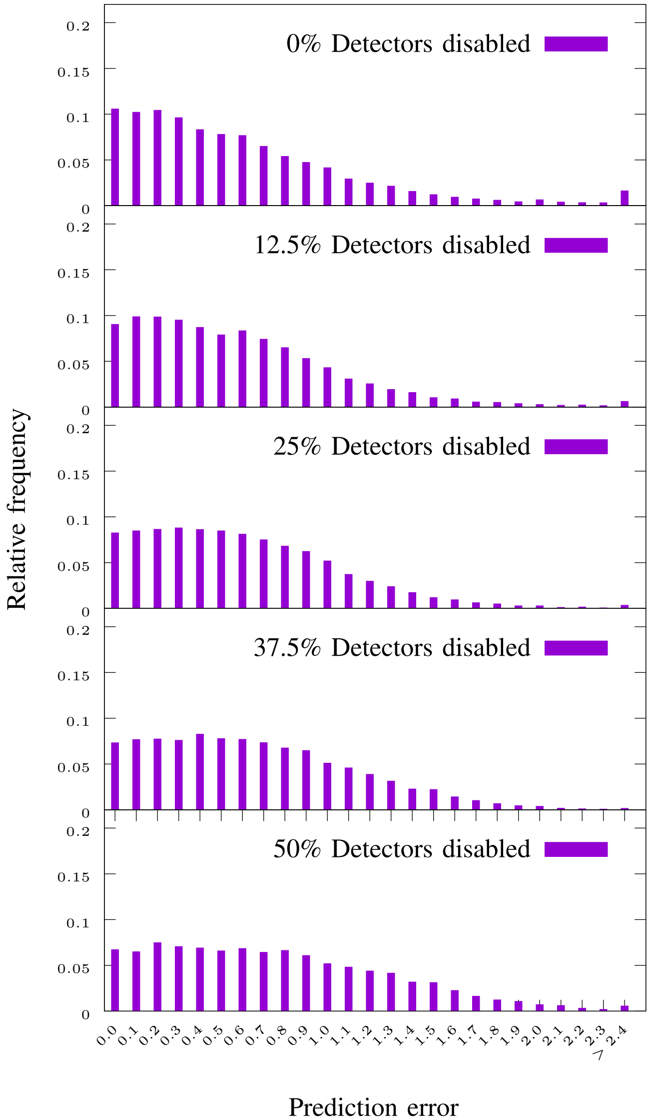


Fig. 4. A histogram of the prediction error of the **continuous feature network** at different numbers of light detectors whose data was withheld.

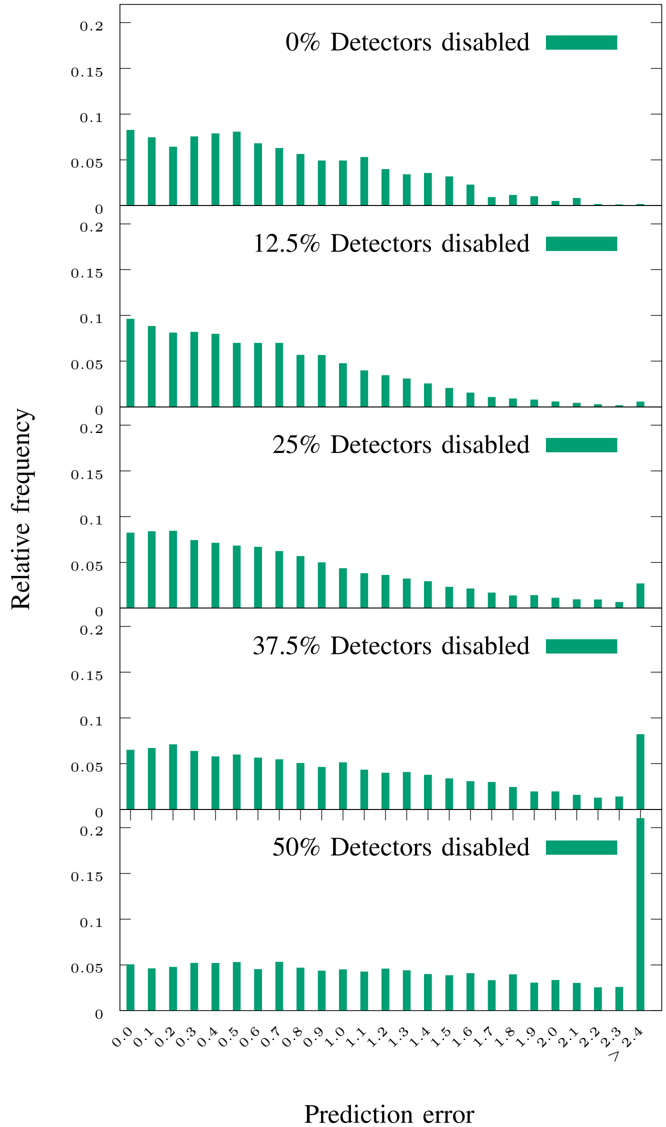


Fig. 5. A histogram of the prediction error of the **multi-layer feed-forward network** at different numbers of light detectors whose data was withheld.

distance. This is needed as the feed-forward network is incapable of handling the shorter input vector without retraining. If no other data was available with a similar wavelength and distance, the value was set to 0 for the feed-forward network.

The results show that the continuous feature network outperforms the similarly-sized multi-layer feed-forward network for all investigated numbers of detectors whose data was withheld. The continuous feature network is able to achieve a mean square error of 19% lower compared to the multi-layer feed-forward network for the full set of input data. The improved prediction accuracy can be explained both by the high suitability of continuous feature networks for MSRRS data allowing an improved abstraction of the relationship between optical data and the reference carotenoid concentration in human skin, as well as because the continuous feature network is able to incorporate the actual measured wavelengths of the light emitters for each sensor as the position of the input data points. In addition, we see that the continuous feature network is able to give a stable prediction with more data missing compared to the multi-layer feed-forward network. While the continuous feature layer without a subsequent network is unable to achieve a similar prediction accuracy compared to the full continuous feature network, it retains its resilience against missing data. The observed mean square error is only slightly increasing with 50% of the data missing. Compared to the multi-layer feed-forward network, the measurement results show that the resilience against missing data is yielded by the use of the continuous kernels in the continuous feature layer for initial data input. Built into a slightly deeper model architecture like the proposed continuous feature network, the continuous kernels in the continuous feature layer are a powerful tool for processing irregularly and inconsistently sampled data.

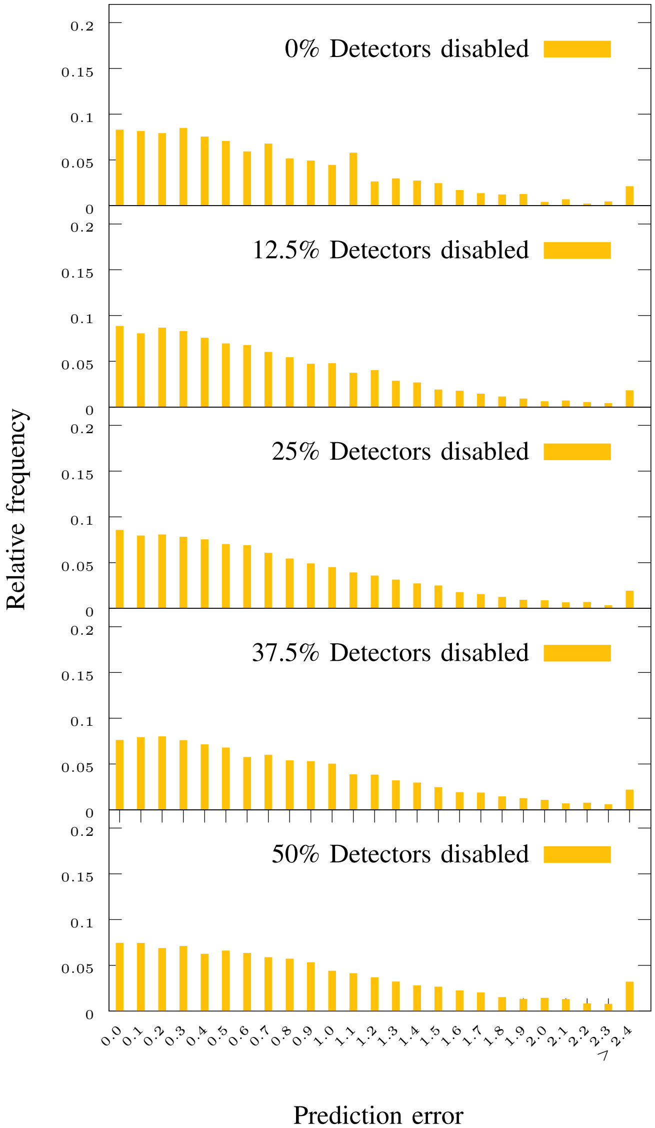
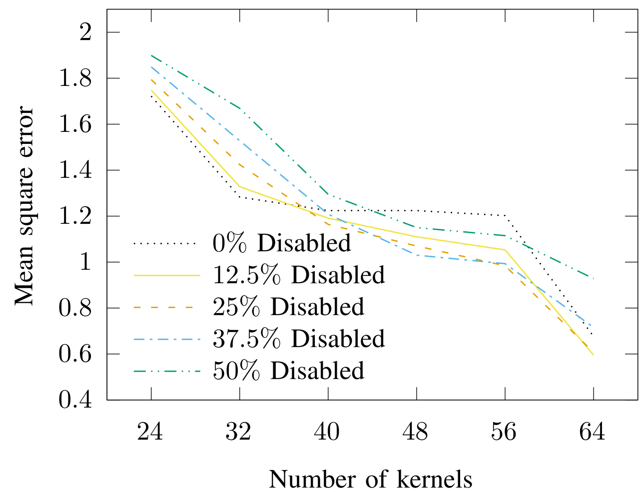


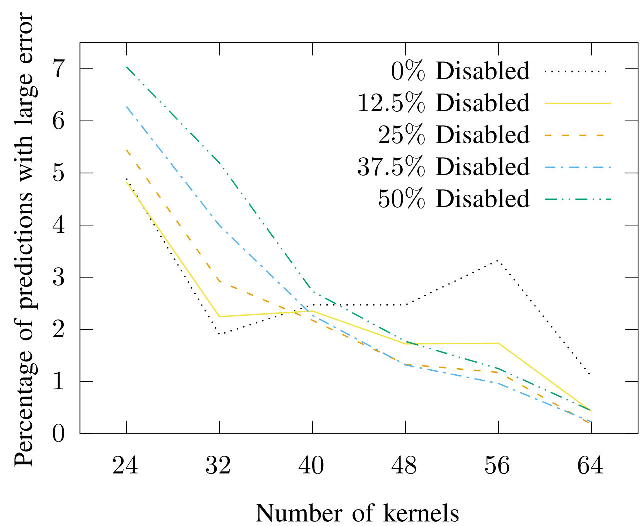
Fig. 6. A histogram of the prediction error of a **continuous feature layer** followed by a simple weighted sum instead of a fully trained follow-up network at different numbers of light detectors whose data was withheld.

The ability of the continuous feature layer and network to adapt to fewer data being available can be seen in more detail in Figures 4, 5, and 6. Figure 4 shows a histogram of the prediction error of the continuous feature network with a different amount of data being withheld. The detectors whose data was withheld from the continuous feature network were picked randomly. As the graph shows, while the amount of highly accurate precision lowers with more data being withheld, the amount of predictions with a large error (> 2.4) is not increasing significantly in contrast to the multi-layer feed-forward network. The multi-layer feed-forward network quickly encounters an increase in predictions with a large error (> 2.4) once the amount of withheld data from disabled detectors increases to or above 25%, in addition to a reduction in highly accurate predictions as seen in Figure 5. This contrast shows that the continuous feature network is capable of adapting to a lower amount of data being available without the need for retraining. The slight increase of predictions with a large error occurring for the continuous feature network when no data is withheld is presumed to be due to a combination of inaccurate input data point density estimation and slight overfitting and will be subject to further investigation.

Figure 6 shows a histogram of the prediction error for the continuous feature layer with no subsequent network. Compared to the full continuous feature network, the continuous feature layer only has less highly accurate predictions when the full amount of data is available. Instead, the number of



(a) The mean square error with predicting carotenoids in human skin using continuous feature networks of different sizes.



(b) The percentage of predictions with a large carotenoids in human skin using continuous feature networks of different sizes.

Fig. 7. The accuracy of continuous feature networks for predicting carotenoids in human skin from optical data, tested for continuous feature layers containing different amounts of continuous kernels and with the data of different percentages of detectors withheld from the continuous feature network.

2023, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

moderately accurate predictions increases, while the number of predictions with a large error slightly increases as well. Similar to the full continuous network, the measurements show that with less and less data being available, the continuous feature layer without subsequent network is also capable of keeping the number of predictions with a large error low, with only a slight increase being observable, compared to the large increase of the multi-layer feed-forward network.

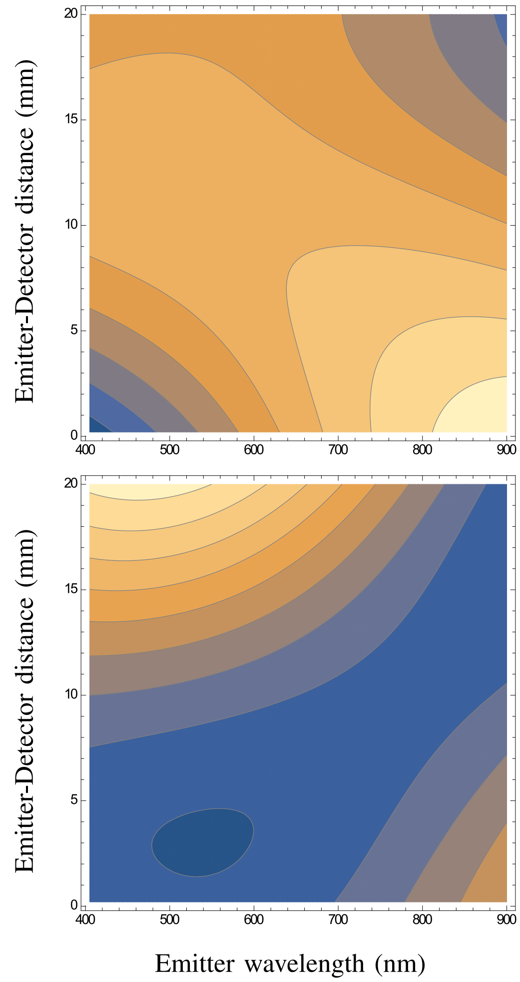
Influence of the Number of Kernels Used

One important parameter of the continuous feature network is the number of continuous kernels used in the continuous feature layer. As can be seen in Figure 7, a larger number of kernels will result in a better prediction of carotenoids in the test dataset. This is visible both in the reducing mean square error shown in Figure 7a, as well as in the reducing number of predictions with a large error (> 2.4) seen in Figure 7b. One interesting observation is that the number of predictions with a large error increases specifically for situations with many kernels used in the network as well as with few, especially no detectors having their data withheld. This supports the assumption that the slight increase in predictions with a large error, as observed before, may be caused by overfitting, especially in the larger networks. For this reason, analysis of the efficacy of anti-overfitting techniques such as dropout [14] for the use with continuous feature layers and continuous feature networks is interesting for future research. Once the network gets sufficiently large, the overall increase in prediction quality is able to outperform the increasing effect of overfitting when all data is available.

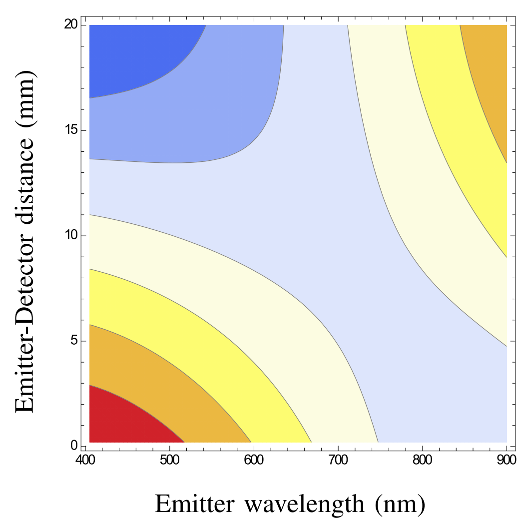
VII. Potential for Explainable AI

A side effect of the continuous feature layer is the resulting potential for explainable AI. As continuous kernels represent weights for each position in the measurement domain, we can deduct levels of importance of certain regions within the measurement domain from the encoded weights. The average of the absolute value of the weights over all kernels might be used as a measure of the importance of the data at certain points in the measurement domain. This may allow the use of the learned continuous kernels as an interpretable model [15].

Figure 8 shows three such kernels. They are taken from the continuous feature layer with no subsequent network, trained on predicting the carotenoid concentration in human skin, as presented in Figures 3 and 6 from Section VI. Figure 8a shows two kernels whose outputs have some of the highest weights in the learned network and thus have some of the highest influence. Figure 8b shows a kernel whose output has a negative weight with one of the highest absolute values. Since the output of this model is directly acquired from a weighted sum of the activated outputs of the continuous feature layer, it is possible to infer meaning directly from the learned kernels. In the positive kernels shown in Figure 8a, a high value in the kernel (here bright yellow/white) means that an emitter-detector pair with the corresponding wavelength and distance will have a high impact, and a high brightness will



(a) Two kernel whose outputs have a large **positive** weight. Yellow areas imply high weight, blue areas imply low weight.



(b) A kernel whose output has a large **negative** weight. Red areas imply high weight, blue areas imply low weight.

Fig. 8. Kernels from a continuous feature layer whose outputs are combined in a weighted sum, trained to predict carotenoids in human skin. High-weight areas in positively weighted kernels increase the final value when more light is detected, while high-weight areas in negatively weighted kernels decrease the final value when more light is detected.

thus increase the final value. At the same time, in the negative kernel Figure 8b, a high value (here red) means that an emitter-detector pair with the corresponding wavelength and distance will have a high impact, and a high brightness will thus decrease the final value. The positive kernels show that more light detected at various wavelengths will increase the final value, except for light at low distances and wavelengths around $400\ \mathrm{nm}$ to $600\ \mathrm{nm}$. Similarly, the negative kernel shows that especially more light being detected at low distances and wavelengths between $400\ \mathrm{nm}$ to $500\ \mathrm{nm}$ will decrease the final value. This is plausible, as the main detection targets are carotenoids. It is known that carotenoids have a high absorption, especially between $450\ \mathrm{nm}$ and $500\ \mathrm{nm}$ [16]. It is thus expected that if more light in this absorption band is reaching the light detectors, fewer carotenoids seem to be present to absorb it, and thus the predicted carotenoid concentration is lowered, as the negative kernel shown in Figure 8b will cause.

This shows that kernel interpretability is an interesting feature of continuous feature layers. Combining kernel interpretability with more tools of explainable AI, especially to enable model interpretability for models with more complex networks following the continuous feature layer, are an interesting topic for future research.

VIII. Conclusion and Future Work

This paper proposes the continuous feature network, a novel method to process irregularly and inconsistently sampled data with position-dependent features, such as optical or acoustic spectra. In addition, the continuous feature network is shown to outperform a comparable multi-layer feed-forward network with a 19% lower mean square error on predicting carotenoid concentration in human skin from optical multiple spatially resolved reflection spectroscopy data. This shows that the continuous feature network performed better at abstracting the relationship between the optical MSRRS data and the reference carotenoid concentration. Furthermore, this paper shows that the continuous feature network is capable of making stable predictions of carotenoid concentration in human skin with up to 50% of the data from the optical detectors withheld, while a comparable multi-layer feed-forward network exhibits a significant increase in predictions with a large error from 25% of the data withheld. Furthermore, it is shown that this reliability is caused by the continuous feature layer itself instead of the subsequent network. Additionally, it is shown how the number of continuous kernels affects the model's resilience and adaptability against missing data, as well as its overall ability to perform accurate predictions of carotenoid concentrations from optical data. Other potential use cases include similar types of data where samples may be irregular and features are position-dependent in the measurement domain, including other types of spectra, such as audio. Additionally, an approach to utilize continuous feature networks as interpretable models for explainable AI is introduced and shown in the example of the prediction of carotenoids.

References

[1] B. M. Magnussen, C. Stern, and B. Sick, "Utilizing continuous kernels for processing irregularly and inconsistently sampled data with position-dependent features," in *Proceedings of The Nineteenth International Conference on Autonomic and Autonomous Systems*, ser. International Conference on Autonomic and Autonomous Systems, C. Behn, Ed., IARIA. ThinkMind, Mar 2023, pp. 49–53.

[2] A. R. T. Donders, G. J. van der Heijden, T. Stijnen, and K. G. Moons, "Review: A gentle introduction to imputation of missing values," *Journal of Clinical Epidemiology*, vol. 59, no. 10, pp. 1087–1091, 2006.

[3] P. K. Sharpe and R. J. Solly, "Dealing with missing values in neural network-based diagnostic systems," *Neural Computing & Applications*, vol. 3, no. 2, pp. 73–77, Jun 1995.

[4] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[5] D. W. Romero, A. Kuzina, E. J. Bekkers, J. M. Tomczak, and M. Hoogendoorn, "Ckconv: Continuous kernel convolution for sequential data," *CoRR*, vol. abs/2102.02611, 2021.

[6] K. T. Schütt, P.-J. Kindermans, H. E. Sauceda, S. Chmiela, A. Tkatchenko, and K.-R. Müller, "Schnet: A continuous-filter convolutional neural network for modeling quantum interactions," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, p. 992–1002.

[7] W. Wu, Z. Qi, and L. Fuxin, "Pointconv: Deep convolutional networks on 3d point clouds," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 9613–9622.

[8] M. E. Darvin, B. Magnussen, J. Lademann, and W. Köcher, "Multiple spatially resolved reflection spectroscopy for in vivo determination of carotenoids in human skin and blood," *Laser Physics Letters*, vol. 13, no. 9, p. 095601, Aug 2016.

[9] V. Sitzmann, J. N. P. Martel, A. W. Bergman, D. B. Lindell, and G. Wetzstein, "Implicit neural representations with periodic activation functions," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020.

[10] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 165–174.

[11] K. Genova, F. Cole, A. Sud, A. Sarna, and T. Funkhouser, "Local deep implicit functions for 3d shape," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2020, pp. 4856–4865.

[12] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.

[13] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019.

[14] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014.

[15] O. Biran and C. V. Cotton, "Explanation and justification in machine learning: A survey," in *IJCAI-17 Workshop on Explainable AI (XAI) Proceedings*, 2017.

[16] E. S. Miller, G. Mackinney, and J. Zscheile, F. P., "Absorption spectra of alpha and beta carotenes and lycopene," *Plant Physiology*, vol. 10, no. 2, pp. 375–381, Apr 1935.

2023, © Copyright by authors, Published under agreement with IARIA - www.iaria.org