THEORY OF FEATURE EXTRACTION

JOY SONG[1]

1. INTRODUCTION

Feature extraction has been a central task nowadays in many machine learning algorithms. In particular, many deep convolutional neural networks (DCNNs) can be interpreted as a process of feature extraction followed by various classifiers. Therefore, developing a mathematical model for feature extraction is of great significance to the analysis of the corresponding algorithms. This paper provides a review for the theory of feature extraction based on deep convolutional neural networks. We introduce the mathematical construction of DCNN-based feature extractors, and summarize the analysis of the properties of these extractors. In the end, we also discuss some possible extensions of the current results.

2. PRELIMINARY

This section introduces some basic concepts and the corresponding notations we are going to use throughout this paper.

2.1. **Deep convolutional neural networks.** A typical deep convolutional neural network propagates the input signals through multiple layers, each of which consists of a series of filters followed by non-linearities and pooling operators. Basically, in each layer the filters serve to extract different aspects of information by convolving with the signal, then the non-linearities and pooling operators are applied for better and faster algorithm performance. A wide range of filters, non-linearities and pooling operators are employed for DCNNs of different purposes.

2.2. **Signals and cartoon functions.** Signals are the input of the DCNN-based feature extractors, and as will be stated later, some properties of the feature extractor also depend on the class of signals it deals with. Here we introduce the mathematical formulation of signals in both continuous and discrete schemes, and also talk about a specific class of signals of interest.

The continuous-time signal is treated as a Lebesgue measurable function $f : \mathbb{R}^d \to \mathbb{C}$, and $L^p(\mathbb{R}^d), 1 \leqslant p < \infty$ denotes the space of Lebesgue measurable functions $f : \mathbb{R}^d \to \mathbb{C}$ satisfying $\|f\|_p := (\int_{\mathbb{R}^d} |f|^p d\lambda^d)^{1/p} < \infty$. In addition, $L^\infty(\mathbb{R}^d)$ stands for the space of Lebesgue measurable functions $f : \mathbb{R}^d \to \mathbb{C}$ satisfying $\|f\|_\infty := \inf\{\alpha > 0 : |f| \leqslant \alpha, a.e. \lambda^d\} < \infty$. The convolution of two signals $f \in L^2(\mathbb{R}^d)$ and $g \in L^1(\mathbb{R}^d)$ is denoted as $(f * g)(y) = \int_{\mathbb{R}^d} f(x)g(y-x)dx$.

The discrete-time signal is treated as a periodic function on the domain $I_N := \{0, 1, \ldots, N-1\}$ and H_N stands for the space of N-periodic functions $f : \mathbb{Z} \to \mathbb{C}$ satisfying $f[n] = f[n+N], \forall n \in \mathbb{Z}$. As an example, with $N = mn$, an input image of size $m \times n$ could be stretched into a vector of

[1]BRIGHAM YOUNG UNIVERSITY
E-mail address: jsong@mathematics.byu.edu.
Date: April 18, 2020.

length N whose i-th entry is $f[i]$ where f is an N-periodic function. Similar to the continuous case, we define $\|f\|_p := (\sum_{n\in I_N} |f[n]|^p)^{1/p}$ for $1 \leqslant p < \infty$ and $\|f\|_\infty := \sup_{n\in I_N} |f[n]|$. The convolution of two signals $f, g \in H_N$ is denoted as $(f * g)[n] := \sum_{k\in I_N} f[k]g[n-k]$.

The function $f : \mathbb{R}^d \to \mathbb{C}$ is defined as a cartoon function [1] if it could be written as $f = f_1 + \mathbb{1}_B f_2$ where $B \subset \mathbb{R}^d$ is a compact Lipschitz domain with boundary $vol^{d-1}(\partial B)$ of finite length, for $i = 1, 2$, $f_i \in L^2(\mathbb{R}^d) \cap C^1(\mathbb{R}^d, \mathbb{C})$ satisfies the decay condition $|\nabla f_i(x)| \leqslant C\langle x\rangle^{-d}$ for some constant $C > 0$, where $\langle x\rangle = (1+|x|^2)^{1/2}$. Furthermore, the *class of cartoon functions of maximal size* $K > 0$ is defined as follows:

$$\mathscr{C}^K_{\text{CART}} := \Big\{ f_1 + \mathbb{1}_B f_2 : f_i \in L^2(\mathbb{R}^d) \cap C^1(\mathbb{R}^d, \mathbb{C}), i = 1, 2,$$
$$|\nabla f_i(x)| \leqslant K\langle x\rangle^{-d}, \text{vol}^{d-1}(\partial B) \leqslant K, \|f_2\|_\infty \leqslant K \Big\}.$$

We are interested in cartoon functions because they are a good model for a large class of input signals in many machine learning algorithms. Interpreted as the superposition of different smooth segments, cartoon functions can represent natural images and images of objects with geometric shapes (e.g., images of handwritten digits).

2.3. **Wavelets.** Similar to Fourier transform, wavelet transform is a widely used method to obtain information encoded in signals. Nowadays in the field of signal processing, wavelet transform is considered even superior due to the fact that it is capable of providing information about time and frequency simultaneously, while Fourier transform can only provide information about frequency.

The *continuous wavelet transform* of a signal $f \in L^2(\mathbb{R}^d)$ is defined as:

$$(W_\psi f)(a, b) = \frac{1}{\sqrt{|a|}} \int \psi^* \left(\frac{t-b}{a} \right) f(t)\, dt.$$

As seen in the above equation, the transformed signal is a function of two variables: the translation parameter b which reads off time information, and the scale parameter a which reads off frequency information. The kernel $\psi(t)$ is called the *mother wavelet* which could be chosen as needed. With different values of the parameters a, b, the dilated and translated wavelets $\frac{1}{\sqrt{|a|}}\psi(\frac{t-b}{a})$ are able to extract any frequency components during any time interval.

As a result, wavelets with customized parameters are used as filters in many deep convolutional neural networks. What is used in [2] is a set of directional wavelets where the mother wavelet ψ is dilated by 2^{-j} and rotated by a rotation matrix r_k in a finite rotation group G:

$$\psi_\lambda(x) := 2^{dj}\psi(2^j r_k^{-1} x), \quad \lambda = (j, k) \in (\mathbb{Z} \times \{0, 1 \ldots, K-1\})$$

They considered a wavelet transform that only keeps wavelets of frequencies $2^j > 2^{-J}$:

$$\{\psi_\lambda\}_{\lambda\in\Lambda_J}, \quad \Lambda_J = \Big\{ \lambda = (j, k) : 2^j > 2^{-J}, j \in \mathbb{Z}, k = 0, 1, \ldots, K-1 \Big\}$$

and covered the low frequencies with a low-pass filter $\psi_{(-J,0)}$ corresponding to the coarsest scale resolved by this set of directional wavelets.

2.4. **Properties of operators.** The feature extractors are expected to be translation-invariant and deformation-stable. In the case of image classification, for example, changing the location (translation) of the central object should not change the classification result. Similarly, we could expect the same output under certain deformations of the object.

As defined in [2], an operator Φ from $L^2(\mathbb{R}^d)$ to a Hilbert space $\mathscr{H}$ is *translation-invariant* if $\Phi(T_t f) = \Phi(f)$ for all $f \in L^2(\mathbb{R}^d)$ and $t \in \mathbb{R}^d$, where $T_t f(x) = f(x-t)$ is the translation of $f \in L^2(\mathbb{R}^d)$ by $t \in \mathbb{R}^d$. It is unrealistic to expect the exact translation-invariance in practice, but under some assumptions, we are able to achieve invariance asymptotically.

An operator Φ is said to be *deformation-stable* if there exists a constant $C > 0$ such that $\|\Phi(F_{\tau,\omega} f) - \Phi(f)\| \leqslant C\|F_{\tau,\omega} f - f\|$ for all $f \in L^2(\mathbb{R}^d)$ where $F_{\tau,\omega}$ denotes a deformation operator. In practice, people developed different deformation sensitivity bounds for $\|\Phi(F_{\tau,\omega} f) - \Phi(f)\|$, dependent on not only the parameters of the deformation operator but also the class of signals we are dealing with.

3. Architecture of DCNNs

A mathematical framework which could be used for DCNNs in a continuous-time case was proposed by Mallat [2] in 2012, where a scattering network is composed of layers of wavelets followed by modulus, and the input signals are propagated through these layers along different paths. Following the "path ordered" idea in the scattering networks, in 2015, Wiatowski and Bölcskei [3] developed a structure specifically for deep convolutional neural networks by adding pooling operators as well as generalizing the filters and non-linearities used in the scattering networks. Based on the previous continuous models, Wiatowski (2016) [4] proposed a simplified model in a discrete scheme for the DCNN-based feature extraction.

3.1. **Scattering network.** According to Mallat [2], for the input signal $f \in L^2(\mathbb{R}^d)$, a *scattering propagator* $U[p]$ is defined to be a path ordered composition of a series of convolution-modulus operators:

$$U[p]f = U[\lambda_m] \cdots U[\lambda_2]U[\lambda_1]f = \big|\cdots\big|\big|f * \psi_{\lambda_1}\big| * \psi_{\lambda_2}\big|\cdots * \psi_{\lambda_m}\big|$$

where $p = (\lambda_1, \lambda_2, \ldots, \lambda_m)$ is an ordered sequence indicating the "path" to choose different wavelets ψ_{λ_k} with parameter $\lambda_k, 1 \leqslant k \leqslant m$. A *windowed scattering transform* $S_J[p]$ is defined to convolve the scattering propagator with the low-pass filter $\psi_{2^{-J}}$ in the end to localize it over spatial domains of size proportional to 2^J:

$$S_J[p]f = \big|\cdots\big|\big|f * \psi_{\lambda_1}\big| * \psi_{\lambda_2}\big|\cdots * \psi_{\lambda_m}\big| * \psi_{2^{-J}}.$$

The *scattering network* is then given by a collection of propagated signals:

$$S_J[\mathscr{P}_J]f = \big\{S_J[p]f\big\}_{p \in \mathscr{P}_J}$$

where the index set $\mathscr{P}_J = \cup_{m \in \mathbb{N}} \Lambda_J^m$ is the set of all finite paths $p = (\lambda_1, \lambda_2, \ldots, \lambda_m), m \in \mathbb{N}$ with parameters $\lambda_k \in \Lambda_J$ as defined in Section 2.3. The architecture of the scattering network is shown in Figure 1.

3.2. **Continuous feature extractor.** Mallat's scattering network provides an important idea to represent layers of operation as a path ordered composition of functions. Generalizing the structure of scattering networks, Wiatowski and Bölcskei [3] proposed a mathematical framework especially designed for deep convolutional neural networks, where each layer now is a triplet composed of filters, non-linearities and pooling operators.

According to [3], a *module sequence* is a sequence of triplets:

$$\Omega = \Big((\Psi_n, M_n, P_n)\Big)_{n \in \mathbb{N}}.$$

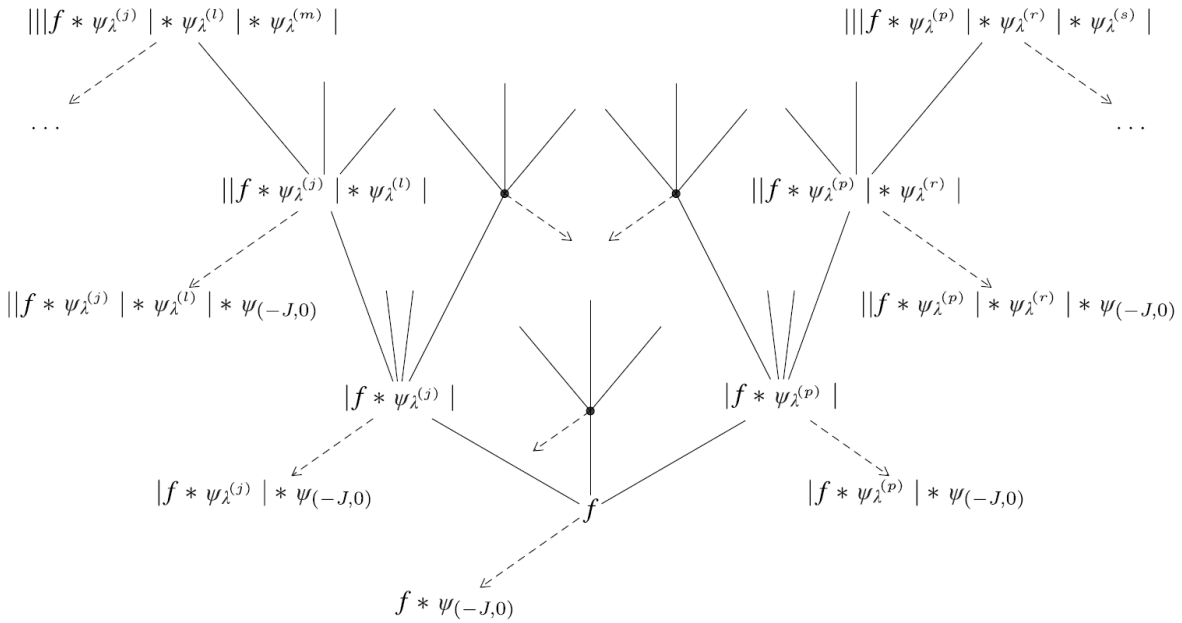

Figure 1. Architecture of the scattering network, taken from [3].

For each $n \in \mathbb{N}$, Ψ_n is a semi-discrete frame associated with a set of atoms $\{g_{\lambda_n}\}_{\lambda_n \in \Lambda_n}$ where $g_{\lambda_n} \in L^1(\mathbb{R}^d) \cap L^2(\mathbb{R}^d)$ are the filters to convolve with, and Λ_n is a countable index set; $M_n, P_n : L^2(\mathbb{R}^d) \to L^2(\mathbb{R}^d)$ are Lipschitz-continuous operators satisfying $M_n f = 0$ and $P_n f = 0$ for $f = 0$. In essence, M_n represents the non-linearity, P_n together with the pooling factor S_n represents the pooling operator.

Based on the module sequence, the operator associated with the n-th layer of the network is given by

$$U_n[\lambda_n]f = S_n^{d/2} P_n(M_n(f * g_{\lambda_n}))(S_n \cdot).$$

Consistent with the definition of a scattering propagator in [2], the *path ordered feature extractor* of the first n layers is then defined as

$$U[q]f := U_n[\lambda_n] \dots U_2[\lambda_2] U_1[\lambda_1] f,$$

where the path $q = (\lambda_1, \lambda_2, \dots, \lambda_n) \in \Lambda_1^n := \Lambda_1 \times \Lambda_2 \cdots \times \Lambda_n$ is an ordered sequence indicating the parameters of filters to be used. For the empty path $e := \varnothing$, they set $\Lambda_1^0 := \{e\}$ and let $U[e]f = f$. Finally, together with an output generator χ_n, we obtain $(U[q]f) * \chi_n$ as an output in the n-th layer generated along the path q.

Therefore, for an input signal $f \in L^2(\mathbb{R}^d)$, the *feature extractor* Φ_Ω based on the module sequence $\Omega = ((\Psi_n, M_n, P_n))_{n \in \mathbb{N}}$ is given by

$$\Phi_\Omega(f) := \cup_{n=0}^{\infty} \Phi_\Omega^n(f),$$

where $\Phi_\Omega^n(f) := \{(U[q]f) * \chi_n\}_{q \in \Lambda_1^n}$ represents the collection of features generated in the n-th layer of the network, and $\Phi_\Omega(f)$ contains all the features generated in every network layer. The architecture of this DCNN-based feature extractor is shown in Figure 2.

3.3. **Discrete feature extractor.** Considering discrete-time signals $f \in H_N$ and finite layers of DCNN network, [4] inherited and further simplified the continuous model proposed in [3].

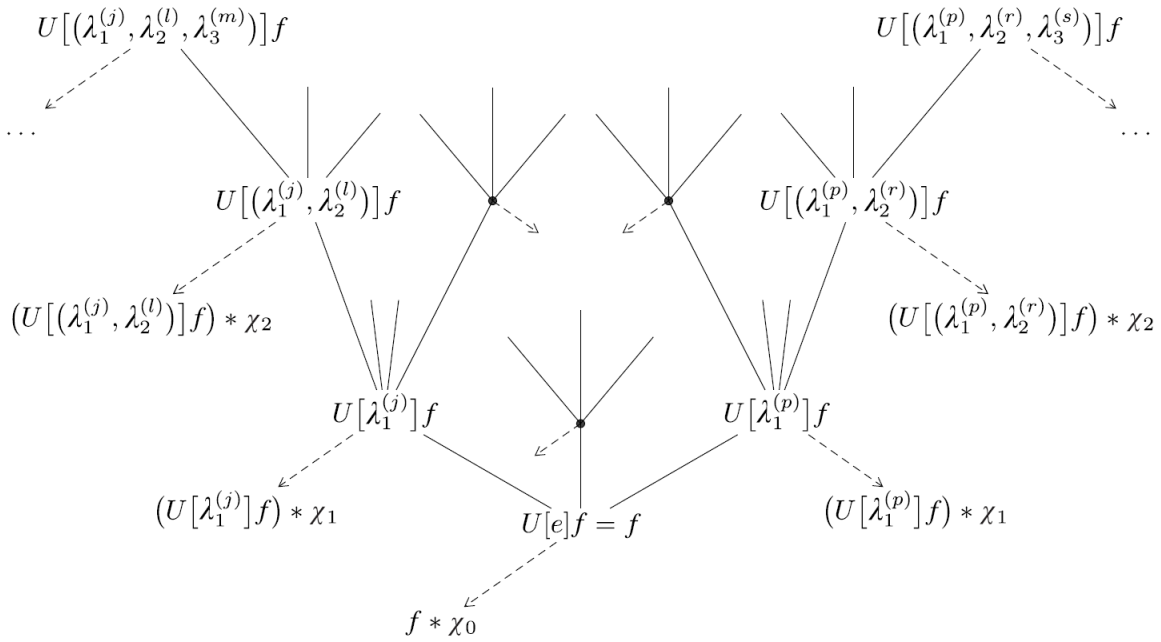

Figure 2. Architecture of the DCNN-based feature extractor, taken from [3].

Here $\Psi_d = \{g_{\lambda_d}\}_{\lambda_d\in\Lambda_d} \subset H_{N_d}$ is a convolutional set, $\rho_d : \mathbb{C} \to \mathbb{C}$ is a point-wise Lipschitz-continuous non-linearity, and $P_d : H_{N_d} \to H_{N_{d+1}}$ is a Lipschitz-continuous pooling operator with $N_{d+1} = N_d/S_d$ where S_d is the pooling factor in the d-th layer. The module sequence Ω is then given by the finite sequence of triplets $((\Psi_d,\rho_d,P_d))_{1\leqslant d\leqslant D}$.

With the same definition of the path $q = (\lambda_1,\lambda_2,\ldots,\lambda_m)$, the path ordered feature extractor $U[q]f$, and the output generator χ_d, they again introduced the feature extractor Φ_Ω based on the module sequence Ω in the discrete setting. Φ_Ω maps $f \in H_{N_1}$ to its features $\Phi_\Omega(f) := \cup_{d=0}^{D-1}\Phi_\Omega^d(f)$, where $\Phi_\Omega^d(f) := \{(U[q]f) * \chi_d\}_{q\in\Lambda_1^d}$ is the collection of features generated in the d-th network layer.

4. Properties of Feature Extractor

As mentioned before, translation-invariance and deformation-stable are the properties of feature extractors which are expected in many real-life applications. For the scattering network, [2] proved asymptotic translation-invariance with respect to the wavelet scale parameter and Lipschitz-continuity to the action of diffeomorphisms. For the feature extractor developed in [3], they obtained translation-invariance with respect to the depth of the network, and they also proposed a deformation sensitivity bound for a specific class of signals. [1] then developed deformation sensitivity bounds for cartoon functions which are more practical than the band-limited functions considered in [3]. Finally in the discrete-time case [4], they treated the previous theorems as the global properties of feature extractors, and further proved similar local properties.

4.1. **translation-invariance.**

Denote the translation by $t \in \mathbb{R}^d$ as T_t, as defined in Section 2.4, an operator Φ is translation-invariant if $\Phi(T_t f) = \Phi(f)$ for all $f \in L^2(\mathbb{R}^d)$ and $t \in \mathbb{R}^d$.

Given the windowed scattering transform $S_J[p]$ and the scattering network $S_J[\mathscr{P}_J]f$ defined in Section 3.1, [2] has proved that

Theorem 1. *For admissible scattering wavelets,*

$$\lim_{J\to\infty} |||S_J[\mathscr{P}_J]f - S_J[\mathscr{P}_J]T_t f||| = 0, \forall f \in L^2(\mathbb{R}^d), \forall t \in \mathbb{R}^d.$$

Proof. See details in [2]. □

This means that the scattering networks achieve translation-invariance asymptotically in the wavelets scaling parameter J. As $J \to \infty$, the wavelets are able to cover the entire range of frequencies, which could be interpreted as if we manage to extract all the information, then translation-invariance could be achieved.

The translation-invariance obtained in [2] only depends on the parameter J, while [3] exhibited a *vertical translation-invariance* of feature extractors which is asymptotic with respect to the network depth:

Theorem 2. *Let $\Omega = ((\Psi_n, M_n, P_n))_{n\in\mathbb{N}}$ be an admissible module sequence, let $S_n \geqslant 1, n \in \mathbb{N}$ denote the pooling factors for P_n, assume that $M_n, P_n : L^2(\mathbb{R}^d) \to L^2(\mathbb{R}^d)$ both commute with the translation operator T_t. If there exists a constant $K > 0$ such that the Fourier transform $\hat{\chi}_n$ of the output generator χ_n satisfies the decay condition: $|\hat{\chi}_n(\omega)||\omega| \leqslant K$, a.e. $\omega \in \mathbb{R}^d, \forall n \in \mathbb{N}$, then for any $f \in L^2(\mathbb{R}^d)$ and any $t \in \mathbb{R}^d$, we have*

$$|||\Phi^n_\Omega(T_t f) - \Phi^n_\Omega(f)||| \leqslant \frac{2\pi|t|K}{S_1 \dots S_n}\|f\|_2$$

Proof. See Appendix F in [3]. □

Here a module sequence $\Omega = ((\Psi_n, M_n, P_n))_{n\in\mathbb{N}}$ is said to be *admissible* if the Bessel bounds $B_n > 0$ of Ψ_n (defined as $\sum_{\lambda\in\Lambda_n} |\hat{g}_\lambda(\omega)|^2 \leqslant B_n$) and the Lipschitz constants $L_n, R_n > 0$ of the operators M_n, P_n satisfy the condition:

$$\max\{B_n, B_n L_n^2 R_n^2\} \leqslant 1, \quad \forall n \in \mathbb{N}.$$

It could be derived from this theorem that

$$\lim_{n\to\infty} |||\Phi^n_\Omega(T_t f) - \Phi^n_\Omega(f)||| = 0, \quad \forall f \in L^2(\mathbb{R}^d), \forall t \in \mathbb{R}^d$$

This means that the DCNN-based feature extractor achieves translation-invariance asymptotically in the number of network layers.

4.2. **deformation-stability.** Consider deformations of the form $(F_{\tau,\omega}f)(x) := e^{2\pi i\omega(x)}f(x-\tau(x))$ where $\omega \in C(\mathbb{R}^d, \mathbb{R})$, and $\tau \in C^1(\mathbb{R}^d, \mathbb{R}^d)$. As defined in Section 2.4, An operator Φ is said to be deformation-stable if there exists a constant $C > 0$ such that $\|\Phi(F_{\tau,\omega}f) - \Phi(f)\| \leqslant C\|F_{\tau,\omega}f - f\|$ for all $f \in L^2(\mathbb{R}^d)$.

For a diffeomorphism action $(F_{\tau,0}f)(x) = f(x-\tau(x))$, [2] developed an upper bound of the difference $|||S_J[\mathscr{P}_J]L_\tau f - S_J[\mathscr{P}_J]f|||$ dependent on the network structure, the signal f itself, and the parameter τ. However, the stability is not guaranteed since there's no knowledge about $\|L_\tau f - f\|$. To establish deformation stability, [3] focused on a specific class of signals: R-band-limited functions $L^2_R(\mathbb{R}^d)$, which satisfies:

$$\|f - F_{\tau,\omega}f\|_2 \leqslant C\Big(R\|\tau\|_\infty + \|\omega\|_\infty\Big)\|f\|_2, \quad \forall f \in L^2_R(\mathbb{R}^d).$$

They then proved the Lipschitz-continuity of the feature extractor Φ_Ω:

Theorem 3. *Let $\Omega = ((\Psi_n, M_n, P_n))_{n\in\mathbb{N}}$ be an admissible module sequence, then the corresponding feature extractor Φ_Ω is Lipschitz-continuous with Lipschitz constant $L_\Omega = 1$, i.e.*

$$|||\Phi_\Omega(f) - \Phi_\Omega(h)||| \leqslant \|f - h\|_2, \quad \forall f, h \in L^2(\mathbb{R}^d).$$

Proof. See Appendix I in [3]. □

The upper bound of $\|f - F_{\tau,\omega}f\|, \forall f \in L^2_R(\mathbb{R}^d)$ combined with the Lipschitz-continuity of Φ_Ω gives us the deformation stability of feature extractors on R-band-limited functions:

$$|||\Phi(F_{\tau,\omega}f) - \Phi(f)||| \leqslant \|F_{\tau,\omega}f - f\| \leqslant C\Big(R\|\tau\|_\infty + \|\omega\|_\infty\Big)\|f\|_2, \quad \forall f \in L^2_R(\mathbb{R}^d).$$

Many signals in practice, however, are not band-limited functions or have a large bandwidth with which this upper bound grows linearly. Therefore, cartoon functions $f \in \mathscr{C}^K_{\text{CART}}$ were considered in [1], which could take structural properties of signals into account. Cartoon functions satisfy $\|f - F_{\tau,0}f\|_2 \leqslant C_K\|\tau\|^{1/2}_\infty$, which combined with the Lipschitz-continuity of the feature extractor again gives an upper bound as follows

$$|||\Phi_\Omega(F_{\tau,0}f) - \Phi_\Omega f||| \leqslant \|F_{\tau,0}f - f\| \leqslant C_K\|\tau\|^{1/2}_\infty, \quad \forall f \in \mathscr{C}^K_{\text{CART}}.$$

Similar deformation sensitivity bound was developed in [4] where they considered discrete input signals and discrete form of cartoon functions.

5. Conclusions

This mathematical construction of the DCNN-based feature extractors is of great significance in that it provides rigorous explanation for both intuitive and empirical results in many of today's machine learning algorithms. Motivated by people's expectation in various classification tasks, the properties of feature extractors developed in [2][3][1][4] verified the robustness of DCNNs under translation and deformation, also verified the validity of expecting better results from deeper networks. Furthermore, a mathematical framework of feature extractors allows us to explore more hidden possibilities to tune different parameters and improve the network.

Some possible extensions of these results lie in different choices of input signals and network operators. (i) In general, the Lipschitz-continuity of the feature extractor Φ_Ω will always guarantee its deformation stability as long as we are able to obtain an upper bound for $\|f - F_{\tau,\omega}f\|$, and this usually requires us to restrict the domain to a certain class of signals. In other words, in a specific problem setting, we can customize the class of signals f we are dealing with and the class of deformations $F_{\tau,\omega}$ we are interested in, then the upper bound we develop for $\|f - F_{\tau,\omega}f\|$ would give us the desired stability of our network. (ii) In addition, the admissibility condition could be loosen to any uniform finite constant upper bound of $\max\{B_n, B_nL^2_nR^2_n\}$. For networks with finite layers, we can even only require layer-wise finite upper bound. (iii) Many commonly used non-linearities (hyperbolic tangent, ReLu, modulus, logistic sigmoid, etc.) and pooling operators (sub-sampling, average-pooling, max-pooling, etc.) are proved to be Lipschitz-continuous [4]. Now having the theorem that the Lipschitz-continuity of the feature extractor could be guaranteed by the Lipschitz-continuity of non-linearities and pooling operators, we are able to develop customized non-linearities and pooling operators as long as they are proved to be Lipschitz-continuous.

REFERENCES

[1] Philipp Grohs, Thomas Wiatowski, and Helmut Bölcskei. Deep convolutional neural networks on cartoon functions. 04 2016.

[2] Stéphane Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65, 10 2012.

[3] Thomas Wiatowski and Helmut Bölcskei. A mathematical theory of deep convolutional neural networks for feature extraction. *IEEE Transactions on Information Theory*, PP, 12 2015.

[4] Thomas Wiatowski, Michael Tschannen, Aleksandar Stanic, Philipp Grohs, and Helmut Bölcskei. Discrete deep feature extraction: A theory and new architectures. 05 2016.