SCATTERING HIDDEN MARKOV TREE

J.B. REGLI, J. D. B. NELSON

UCL, Department of statistical science

ABSTRACT

A Scattering convolutional hidden Markov tree proposes a new inference mechanism for high-dimensional signals by combining the interesting signal representation created by the scattering transform to a powerful probabilistic graphical model. A wavelet scattering network computes a signal translation invariant and stable to deformations representation that still preserves the informative content of the signal. Such properties are acquired by cascading wavelet transform convolutions with non-linear modulus and averaging operators. The network's structure and its distributions are described using a Hidden Markov Tree. This yield a generative model for high-dimensional inference. It offers a mean for performing several inference tasks among which are predictions. The scattering convolutional hidden Markov tree displays promising results on both classification and segmentation tasks of complex images.

Index Terms— Scattering network, Hidden Markov Model, Classification, Deep network

1 Introduction

The standard approach to classify high dimensional signals can be expressed as a two step procedure. First the data are projected in a feature space where the task at hand is simplified. Then prediction is done using a simple predictor in this new representational space. The mapping can either be hand-built —e.g. Fourier transform, wavelet transform— or learned. In the last decade methods for learning the projection have drastically improved under the impulsion of the so called deep learning. Deep neural networks (sometime enriched by convolutional architecture) have been able to learn very effective representations for a given dataset and a given task [**??**]. Such method have achieved state of the art on many standard problems [**?**] as well as real world applications [**?**].

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla pretium posuere auctor. Duis fermentum risus sit amet leo pretium dapibus. Integer a ligula iaculis, suscipit purus nec, malesuada dui. Morbi maximus mattis dolor, sed aliquam diam fermentum at. Fusce fringilla elementum pretium. Sed luctus nulla sit amet est tincidunt pellentesque volutpat id odio. Aenean accumsan ipsum eget velit feugiat, id rhoncus metus aliquet. Fusce vestibulum est id lorem ullamcorper, at iaculis tortor porttitor. Maecenas at ex nec tortor auctor efficitur.

Etiam feugiat vel arcu eget efficitur. Quisque sed nisi nec nunc euismod sodales id eu elit. Sed dapibus interdum felis ut elementum. Maecenas vitae lacus commodo, consectetur nibh a, congue orci. Sed sit amet ipsum semper, imperdiet dolor vitae, pulvinar velit. Sed lorem massa, ornare eu sagittis in, vulputate ac nisl. In vel eros vel neque viverra mollis in sed velit. Nunc finibus magna nunc, eleifend vulputate ipsum iaculis non. Donec mollis sit amet diam ac vehicula. Vestibulum id enim mattis, ornare sapien non, mattis massa. Nulla quis gravida nisl. Integer varius dolor congue sem sagittis, non fermentum massa egestas. Nulla tempus aliquam sapien, in ultricies augue aliquam ac. Nunc et tincidunt felis, eget auctor odio. Aliquam posuere ut leo ut gravida. Fusce malesuada nisi eget tellus pulvinar, sed vulputate enim elementum.

Sed rhoncus, ex vel vulputate eleifend, massa nisl egestas metus, a scelerisque nibh diam vel turpis. Nam pharetra arcu nec rutrum ullamcorper. Aenean lectus odio, mattis in efficitur eget, porttitor et sapien. Cras egestas libero nec nulla faucibus faucibus. Etiam vel enim venenatis mi aliquam efficitur at sit amet erat. Duis et aliquet ipsum, a elementum nisi. Aliquam convallis ligula odio, at bibendum libero tincidunt viverra. Vivamus bibendum molestie tortor, id varius eros laoreet ac. Curabitur quis neque ac erat elementum malesuada eget eget sem. Fusce vulputate nulla sit amet dolor convallis gravida. Nunc rutrum ultrices mi ac pellentesque. Mauris scelerisque pharetra sapien. Donec ultricies ex at neque blandit, finibus tempor libero eleifend. In at tellus diam. Morbi placerat magna nec dignissim sodales. Vivamus aliquam magna est, sed vehicula dolor dictum in.

In Section **??** we present the requisite background of high dimensional signal classification. Section 2 introduces the Scattering Transform and some of its properties. We fuse these to an hidden Markov Tree concepts in Section 3, propose our Scattering Hidden Markov Tree (SCHMT), and describe the inferential machinery. In Section 4 we perform classification on a selection of standard datasets. We draw conclusions in Section **??**

Thanks DSTL/UCL Impact studentship for funding.

2 Scattering networks

Scattering convolutional networks (SCNs) [**?**] are Convolutional Neural Networks (CNNs) **?** using a fixed filter bank of wavelets. Those filters can be hand-crafted to yield descriptors with the desired invariances [**???**]. For image classification tasks, one is interested in descriptors that are —at least— stable to deformations and invariant to translations. Note that SCNs producing more complexes set of invariances exist but on the remainder of this paper we consider only on descriptors with the previously mentioned properties.

2.1 Scattering transform

Wavelets are localized functions stable to deformations. They are thus well adapted to construct descriptor that would also be translation invariant. A two-dimensional spatial wavelet transform W is obtained by scaling by 2^j and rotating by r_θ a mother wavelet ψ,

$$\psi_\lambda(u) = \psi_{j,\theta}(u) = 2^{-2j}\psi(2^{-j}r_\theta p) \quad (1)$$

In the remainder of this paper we restrict to Morlet wavelet transforms defined on $\Lambda = G \times [\![0, J]\!]$ where G is a finite group of rotations of cardinal L and where the wavelet is taken at scale J,

$$W_J\mathbf{x} = \{\mathbf{x} * \phi_J(u); \mathbf{x} * \psi_\lambda(u)\}_{p\in\mathbb{R}^2,\lambda\in\Lambda} \quad (2)$$

While the averaging part ϕ_J of the wavelet transform is invariant to translations, the high frequency part ψ_λ is covariant to them [**?**]. Invariance within a limited range inferior 2^J can be achieved by averaging the positive envelope with a smooth window,

$$S_J[\lambda]\mathbf{x}(u) = |\mathbf{x} * \psi_\lambda| * \phi_J(u) \quad (3)$$

Such non-linearised averaged wavelet coefficients are used in various form in computer vision (SIFT [**?**], DAISY [**?**]), but the scattering transform proposes a new non-linearity as well as a layer based architecture.

2.2 Scattering convolutional network

While providing local translation invariance, the averaging convolution introduced in 3 also removes the spatial variability of the wavelet transform. SCNs cascade this wavelet modulus operator to recover the lost information and compute progressively more invariant descriptors. Let combine the wavelet transform and modulus operations into a single wavelet modulus operator,

$$\mathcal{U}_J\mathbf{x} = \{S[\emptyset]\mathbf{x}; U[\lambda]\mathbf{x}\}_{\lambda\in\Lambda_J} = \{\mathbf{x}*\phi_J; |\mathbf{x} * \psi_\lambda|\}_{\lambda\in\Lambda_J}, \quad (4)$$

A scattering transform can be interpreted as a CNN illustrated in Figure 1 [**?**] which propagates a signal $\mathbf{x}$ across multiple layers of the network and which outputs at each layer m the scattering invariant coefficients $S[p_m]\mathbf{x}$ where $p_m = (\lambda_1 \dots \lambda_m)$ is a path of m orientations and scales.

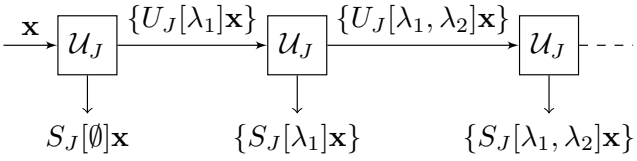

Fig. 1. Scattering networks can be seen as neural networks iterating over wavelet modulus operators $\mathcal{U}_j$. Each layer m outputs the averaged invariants $S[p_m]\mathbf{x}$ and covariant coefficients $U[p_{m+1}]\mathbf{x}$.

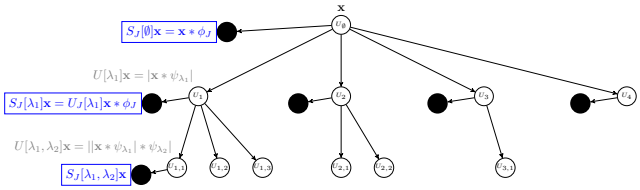

Fig. 2. Frequency decreasing scattering convolution network with $J = 4$, $L = 1$ and $M = 2$. A node i at scale j_i generates $(j_i - 1) \times L$ nodes.

The scattering energy is mainly concentrated along frequency decreasing paths, i.e. for which $|\lambda_{k+1}| \leq |\lambda_k|$ [**?**]. The energy contained in the other paths is negligible and thus for applications only frequency decreasing paths are considered. Moreover there exist a path length $M > 0$ after which all longer paths can be neglected. For signal processing applications, this decay appears to be exponential. And for classification applications, paths of length $M = 3$, i.e. two convolutions, provides the most interesting results [**?**], [**?**].

This restrictions yield an easier parametrization of a scattering network. Indeed its now completely defined by the mother wavelet ϕ, the maximum path length considered M, the finest scale level considered J and the number of orientation considered L.

Hence for a given set of parameter (ψ, M, J, L), let $ST_{(\psi,M,J,L)}(\mathbf{x})$ denotes the unique frequency decreasing windowed scattering convolutional network with those parameters evaluated for signal $\mathbf{x}$. Each node i of this network generates a -possibly empty- set of of nodes of size $(j_i - 1) \times L$ where j_i is the scale of node i and L is the number of orientations considered and it has the architecture displayed by Figure 2.

2.3 Scattering convolutional classifier

In the original framework, the scattering network $ST_{(\psi,M,J,L)}(.)$ is used for classification task using a SVM classifier on the outputs of the network. Performance can be slightly improved by adding a feature selection step performing PCA on the scattering coefficients and keeping only the most informative ones. This classification framework provides results comparable with the state of the art on several datasets [**?**].

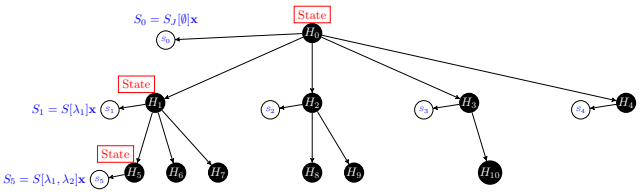

Fig. 3. Scattering convolutional hidden Markov tree.

3 The Scattering hidden Markov tree

State of the art performance can be achieved using SCNs associated to SVMs. However this approach is not adapted to very small training sets or to deliver a probabilistic output. To overcome those limitations we propose an adaptation of Crouse [**?**] and Durand [**?**] wavelet hidden Markov trees to the non-regular non-homogeneous tree structure of SCNs.

3.1 Hidden Markov tree model

The HMT models the marginal distribution of each real ST coefficient S_i as a Gaussian mixture. To each S_i, we associate a discrete hidden state H_i that takes on values in $[\![1, K]\!]$ with probability mass function (pmf) $P(H_i)$. Conditioned on $H_i = k$, S_i is Gaussian with mean $\mu_{i,k}$ and variance $\sigma_{i,k}$. Thus, its overall marginal PDF is given by $P(w_i) = \sum_{k=1}^{K} P(H_i = m)P(S_i|H_i = k)$ with $P(S_i|H_i = k) \sim \mathcal{N}(\mu_{i,k}, \sigma_{i,k})$. While each scattering coefficient S_i is conditionally gaussian given its state variable H_i, overall it has a non-Gaussian density. Finally the probability for the hidden node H_i to be in a state k given its father's state g is characterized by a transition probability such that $\epsilon_i^{(gk)} = P(H_i = k|H_{\rho(i)} = g)$. This yields $P(H_i = k) = \sum_{g=1}^{K} \epsilon_i^{(gk)} P(H_{\rho(i)} = g)$.

Such a model is pictured in Figure 3 and for a given scattering architecture —i.e. fixed M, J and L— the SCHMT model is fully parametrized by,

$$\Theta = \left(\pi_0, \{\epsilon_i, \{\theta_{k,i}\}_{k\in[\![1,K]\!]}\}_{i\in\mathcal{T}}\right). \tag{5}$$

This model implies two assumptions on the scattering transform. First — K-populations— that a signal's scattering coefficients can be described by K clusters. This is a common assumptions for standard wavelets [**?**] and hence it can be extended to the scattering transform. The SCHMT also assumed —persistence— that the informative character of a coefficients is propagated across layers. This assumption is sound since scattering coefficients are highly correlated [**?**].

3.2 Learning the tree parameters

The SCHMT is trained using the smoothed version of the Expectation-Maximization (EM) algorithm [**?**] for hidden Markov trees proposed by [**?**] and adapted to non-homogeneous and non-binary trees.

Let $\bar{\mathcal{S}}_i = \bar{s}_i$ be the observed sub-tree rooted at node i. By convention $\bar{\mathcal{S}}_0$ denotes the entire observed tree. The smoothed version of the E-step requires the computation of the conditional probability distributions $\xi_i(k) = P(H_i = k|\bar{\mathcal{S}}_i = \bar{s}_i)$ (smoothed probability) and $P(H_i = k, H_{\rho(i)} = g|\bar{\mathcal{S}}_i = \bar{s}_i)$ for each node $i \in \mathcal{T}$ and states k and g. This can be achieved through an upward-downward recursion displayed in Algorithm 1 and 2.

// Initialization :
for *All the nodes i of the tree $\mathcal{T}$* **do**
  $P_{\theta_{k,i}}(s_i) = \mathcal{N}(s_i|\mu_{k,i}, \sigma_{k,i})$
end
for *All the leaves i of the tree $\mathcal{T}$* **do**
  $\beta_i(k) = \frac{P_{\theta_{k,i}}(s_i)P(H_i=k)}{\sum_{g=1}^{K} P_{\theta_{g,i}}(s_i)P(H_i=g)}$
  $\beta_{i,\rho(i)}(k) = \sum_{g=1}^{K} \frac{\beta_i(g)\epsilon_i^{(kg)}}{P(H_i=g)}.P(H_{\rho(i)} = k)$
  $l_i = 0$
end
// Induction :
for *All non-leaf nodes i of the tree $\mathcal{T}$ (Bottom-up)* **do**
  $M_i = \sum_{k=1}^{K} P_{\theta_{k,i}}(s_i) \prod_{j\in c(i)} \frac{\beta_{j,i}(k)}{P(H_i=k)^{n_i-1}}$
  $l_i = \log(M_i) + \sum_{j\in c(i)} l_j$
  $\beta_i(k) = \frac{P_{\theta_{k,i}(s_i)} \prod_{j\in c(i)}(\beta_{j,i}(k))}{P(H_i=k)^{n_i-1}M_i}$
  for *All the children nodes j of node i* **do**
    $\beta_{i\setminus c(i)}(k) = \frac{\beta_i(k)}{\beta_{i,j}(k)}$
  end
  $\beta_{i,\rho(i)}(k) = \sum_{g=1}^{K} \frac{\beta_i(g)\epsilon_i^{(kg)}}{P(H_i=g)}.P(H_{\rho(i)} = k)$
end

Algorithm 1: Smoothed upward algorithm.

// Initialization :
$\alpha_0(k) = 1$
// Induction :
for *All nodes i of the tree $\mathcal{T}\setminus\{0\}$ (Top-Down)* **do**
  $\alpha_i(k) = \frac{1}{P(H_i=k)} \sum_{g=1}^{K} \alpha_{\rho(i)}(g)\epsilon_i^{(gk)}\beta_{\rho(i)\setminus i}(g)P(H_{\rho(i)} = g)$
end

Algorithm 2: Smoothed downward algorithm.

// Initialization :
$\pi_0(k) = \frac{1}{N}\sum_{n=1}^{N} P(H_0^n = m|s_0^n, \Theta^l)$
// Induction :
for *All nodes i of the tree $\mathcal{T}\setminus\{0\}$* **do**
  $P(H_i = k) = \frac{1}{N}\sum_{n=1}^{N} P(H_i^n = k|\bar{s}_0^n, \Theta^l)$,
  $\epsilon_i^{gk} = \frac{\sum_{n=1}^{N} P(H_i^n=k, H_{\rho(i)}^n=g|\bar{s}_0^n,\Theta^l)}{NP(H_{\rho(i)}=k)}$,
  $\mu_{k,i} = \frac{\sum_{n=1}^{N} s_i^n P(H_i^n=k|\bar{s}_0^n,\Theta^l)}{NP(H_i=k)}$,
  $\sigma_{k,i}^2 = \frac{\sum_{n=1}^{N}(s_i^n-\mu_{k,i})^2 P(H_i^n=k|\bar{s}_0^n,\Theta^l)}{NP(H_i=k)}$.
end

Algorithm 3: M-step of the EM algorithm.

3.3 MAP classification

Let Θ_c now be a set of parameters for an SCHMT $\mathcal{T}$ learned on a training set $\{\bar{S}_{0,c}^n\}_{n\in[\![1,N]\!]} = \{ST_{(\psi,J,M,L)}(\mathbf{x}_c^n)\}_{n\in[\![1,N]\!]}$ composed of the scattering representations of N realizations of a signal of class c . Let also $\mathbf{x}^{new}$ be another realization of

Fig. 4. add description

this signal, not used for training and $\mathcal{T}^{new}$ be the instance of the SCHMT generated by this realization.

In this context the MAP algorithm aims at finding the optimal hidden tree $\hat{\bar{h}}_0^{new} = (\hat{h}_0^{new} \dots \hat{h}_{I-1}^{new})$ maximizing the probability of this sequence given the model's parameters $P(\bar{\mathcal{H}}_0 = \hat{\bar{h}}_0^{new} | \mathcal{T}^{new}, \Theta_c)$. The MAP framework also provides $\hat{P}$ the value of this maximum.

The MAP Algorithm can be used in a multi-class classification problem by training an SCHMT model per class and then when presented with a new realization $\mathbf{x}^{new}$ comparing the probability of the MAP hidden tree provided by each model

4 Classification results

We compare the performance of SCHMT to those of a SCN combined to an SVM (SCN+SVM) on restrictions to small number of training examples of two standard datasets.

4.1 MNIST

We train both SCHMT and SCN+SVM 100 times on a limited number of training examples per class. The results are displayed in Figure 4

Average results for low number of training examples —2 and 5— are slightly higher for SCHMT compare to SCN+SVM. However the EM algorithm performances are still undermined by convergence issues. When convergence occurs correctly SCHMTs reach better performances than the best SVMs — 80% compared to 54% for 5 training examples. When the number of training samples grow large enough for the SVMs, this method reaches both better maximum and average experiments. Finally we test the best SCHMT and SCN+SVM models trained on 5 images per class on the full testing set.

When convergence occurs correctly SCHMT provides great classification score for low number of training examples

4.2 KTH Texture

5 Conclusion

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc dictum venenatis finibus. Duis faucibus scelerisque hendrerit. Duis ex eros, auctor non massa non, pretium congue magna. Etiam feugiat urna in tortor consequat mattis. Nulla condimentum in arcu nec euismod. Donec ut luctus est. Mauris pellentesque, lectus ornare luctus tincidunt, lorem leo ultrices nunc, efficitur suscipit nunc mauris at ipsum. Donec eros nibh, rhoncus at mi id, mattis varius magna. Nullam luctus nisl nisl, sit amet tincidunt turpis venenatis et. Nam sem lectus, feugiat eget nisl a, dapibus vulputate urna. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Morbi tincidunt aliquet dolor, eu efficitur tortor tempus id. Quisque vestibulum sapien non metus finibus, ut eleifend leo molestie.

Sed ultricies lorem sapien, non sodales ligula bibendum vel. Mauris nec elementum elit. Cras at massa nec nisi dictum bibendum. In commodo vestibulum dapibus. Aenean efficitur, tellus non imperdiet.

6 COPYRIGHT FORMS

ł1abelsec :copyright

You must include your fully completed, signed IEEE copyright release form when form when you submit your paper. We **must** have this form before your paper can be published in the proceedings.

7 REFERENCES