IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium | 979-8-3503-6032-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/IGARSS53475.2024.10641831

Physics-informed Machine Learning for Deep Ice Layer Tracing in SAR images

Maryam Rahnemoonfar
Department of Computer Science and Engineering
Department of Civil and Environmental Engineering
Lehigh University
Bethlehem, PA, USA
maryam@lehigh.edu

Benjamin Zalatan
Department of Computer Science and Engineering
Lehigh University
Bethlehem, PA, USA
bjz222@lehigh.edu

***Abstract*—The precise prediction and tracking of deep internal layers of ice sheets is becoming increasingly important as we deal with the impacts of climate change and the rise of global atmospheric temperatures. Synthetic Aperture Radar (SAR) is the only sensor capable of penetrating through ice and providing us with information about what lies beneath the ice surface, allowing us to monitor changes on a large scale. Forecasting and tracking these internal ice sheet layers is crucial for calculating snow mass balance, inferring otherwise difficult-to-observe ice dynamic processes, and extrapolating ice age from direct measurements of the subsurface. To achieve this, we developed a geometric deep learning model that uses a supervised, multi-target, adaptive long short-term memory graph convolutional network to predict the thicknesses of multiple deep ice layers at specific coordinates in an ice sheet given the thicknesses of a few shallow ice layers. Furthermore, we expanded the model to consider additional physical features of the ice, alongside layer thickness. We found that the inclusion of snow mass balance, meltwater refreezing, and height change due to melting as node features give our model better and more consistent performance.**

***Index Terms*—Deep learning, graph neural networks, recurrent neural networks, airborne radar, ice thickness**

I. Introduction

With the rise in global atmospheric temperatures and changing climate patterns, accurate tracking and prediction of polar snow accumulation and ice thickness has become increasingly important. Understanding the variability of polar snow accumulation over time and space is crucial for reducing uncertainties in climate model predictions, particularly sea level rise. This is achieved by studying the internal ice layers of polar ice sheets, which represent annual isochrones and provide information about the climate of that location during the corresponding year. Tracking and forecasting these internal ice layers is also essential for calculating snow mass balance, extrapolating ice age, and inferring other difficult-to-observe processes.

Traditionally, to measure the ice thickness, ice cores and shallow pits are drilled. However, it is exceedingly difficult to capture catchment-wide accumulation rates using these methods due to their inherent sparsity, access difficulty, high cost, and depth limitations. Attempts to interpolate these in-situ measurements introduce further uncertainties to climate models, especially considering the high variability in local accumulation rate.

Airborne measurements using nadir-looking Synthetic Aperture Radar (SAR) sensors has quickly become a popular complementary method of mapping ice sheet topography and monitoring accumulation rates with a broad spatial coverage and ability to penetrate deep ice layers. The Center for Remote Sensing of Ice Sheets (CReSIS), as part of NASA's Operation Ice Bridge, operates the Snow Radar [1], an airborne SAR sensor that takes high-resolution echograms of polar ice sheets.

Recent studies involving graph convolutional networks (GCNs) [2] have shown promise in spatiotemporal tasks such as traffic forecasting [3]–[5], wind speed forecasting [6], and power outage prediction [7]. In a previous study, we proposed a geometric deep learning model that uses a supervised, multi-target, adaptive long short-term memory graph convolutional network (AGCN-LSTM) [8], [9] to predict the thicknesses of multiple deep ice layers at specific coordinates in an ice sheet given the thicknesses of few shallow ice layers. In this paper, we expand the model to consider additional physical features of the ice, alongside layer thickness. More specifically, we analyze performance when using combinations of each ice layers' snow mass balance (SMB), surface temperature, meltwater refreezing, height change due to melt, and snowpack height as node features.

In our experiments, we use a sample of Snow Radar flights over Greenland in the year 2012. We convert this internal ice layer data into sequences of temporal graphs to be used as input to our model. We convert the five shallow ice layers beneath the surface into five spatiotemporal graphs. Our model then performs multi-target regression to predict the thicknesses of the fifteen deep ice layers beneath them. Our modified model was shown to perform better and with more consistency than previous models.

II. Related Work

A. Automated Ice Layer Segmentation

In recent years, automated techniques have been developed to track the surface and bottom layers of an ice sheet using radar depth sounder sensors. Tracking the internal layers, however, is more difficult due to the low proximity between each layer, as well as the high amount of noise present in the echogram images. Due to its exceptional performance in automatic feature extraction and image segmentation tasks, deep learning has been applied extensively on ice sheet echograms in order to track their internal layers [10]–[13]. [12] used a multi-scale contour-detection convolutional neural network (CNN) to segment the different internal ice layers within Snow Radar echogram images. In [10], the authors trained a multi-scale neural network on synthetic Snow Radar images for more robust training. A multi-scale network was also used in [13], where the authors trained a model on echograms taken in the year 2012 and then fine tuned it by training on a small number of echograms taken in other years. [11] found that using pyramid pooling modules, a type of multi-scale architecture, helps in learning the spatio-contextual distribution of pixels for a certain ice layer. The authors also found that denoising the input images improved both the model's accuracy and F-score. While these models have attempted to segment Snow Radar echogram images, none have yet attempted to predict deep ice layer thicknesses with only information about shallow ice layers.

B. Graph Convolutional Networks

Graph convolutional networks have had a number of applications in a vast array of different fields. In the field of computer vision, recurrent GCNs have been used to generate and refine "scene graphs", in which each node corresponds to the bounding box of an object in an image and the edges between nodes are weighted by a learned "relatedness" factor [14], [15]. GCNs have also been used to segment and classify point clouds generated from LiDAR scans [16], [17]. Recurrent GCNs have been used in traffic forecasting, such as in [3], where graph nodes represented traffic sensors, edges were weighted by the physical distance between sensors, and node features consisted of the average detected traffic speed over some period of time.

Some existing graph-based weather prediction models, such as [6] and [18], have tested models in which edge weights are defined as learnable parameters rather than static values. This strategy allowed the models to learn relationships between nodes more complex than simple geographic distance, and was shown to improve performance at the expense of increased computational complexity.

In a previous study [19], we used a GCN-LSTM to predict the thicknesses of shallow ice layers using the thicknesses of deep ice layers. Our results were reasonable, usually lying within 5 pixels of the ground-truth, and we found that GCN-LSTM performed better and with more consistency than equivalent non-temporal and non-geometric models. In a follow-up paper [20], we modified the model structure, most notably implementing an adaptive layer prior to the GCN-LSTM layer. This modified structure proved to give even better results, though remained simplistic in its node features, using only latitiude, longitude, and ice layer thickness.

III. Dataset

In this study, we use the Snow Radar dataset made public by CReSIS as part of NASA's Operation Ice Bridge. The Snow Radar operates from 2-8 GHz and is able to track deep ice layers with a high resolution over wide areas of an ice sheet. The sensor produces a two-dimensional grayscale profile of historic snow accumulation over consecutive years, where the horizontal axis represents the along-track direction, and the vertical axis represents layer depth. Pixel brightness is directly proportional to the strength of the returning signal. Each of these grayscale echogram profiles has a width of 256 pixels and a height ranging between 1200 and 1700 pixels. Each pixel in a column corresponds to approximately 4cm of ice, and each echogram image has an along-track footprint of 14.5m. Accompanying each image are vectors that provide positional data (including geographic latitude and longitude) of the sensor for each column, as well as Model Regional Atmospheric (MAR) physical data for each of the ice layers. Snow mass balance (SMB), surface temperature, meltwater refreezing, height change due to melt, and snowpack height were extracted from MAR model. In order to gather ground-truth thickness data, the images were manually labelled in a binary format where white pixels represented the tops of each firn layer, and all other pixels were black. Thickness data was extracted by finding the distance (in pixels) between each white pixel in a vertical column.

We focus on radar data captured over Greenland during the year 2012. Since each ice layer often represents an annual isochrone, we may refer to specific layers by their corresponding year (in this case, the surface layer corresponds with the year 2012, the layer below it 2011, and so on). In order to capture a sufficient amount of data, only echogram images containing a miniumum of 20 ice layers were used (five feature layers and fifteen predicted layers). Five and fifteen feature and predicted layers, respectively, were chosen in order to maximize the number of usable images while maintaining a sufficient number of experimental layers. This restriction reduced the total number of usable images down to 703. Five different training and testing sets were generated by taking five random permutations of all usable images and splitting them at a ratio of 4:1. Each training set contained 562 images, and each testing set contained 141 images.

IV. Methods

A. Graph Convolutional Networks

Traditional convolutional neural networks use a matrix of learnable weights, often referred to as a kernel or filter, as a sliding window across pixels in an input image. The result is a higher-dimensional representation of the image that automatically extracts image features that would otherwise need to be identified and inputted manually. Graph convolutional networks apply similar logic to graphs, but rather than using a sliding window of learned weights across a matrix of pixels, GCN performs weighted-average convolution on each node's neighborhood to automatically extract features that reflect the

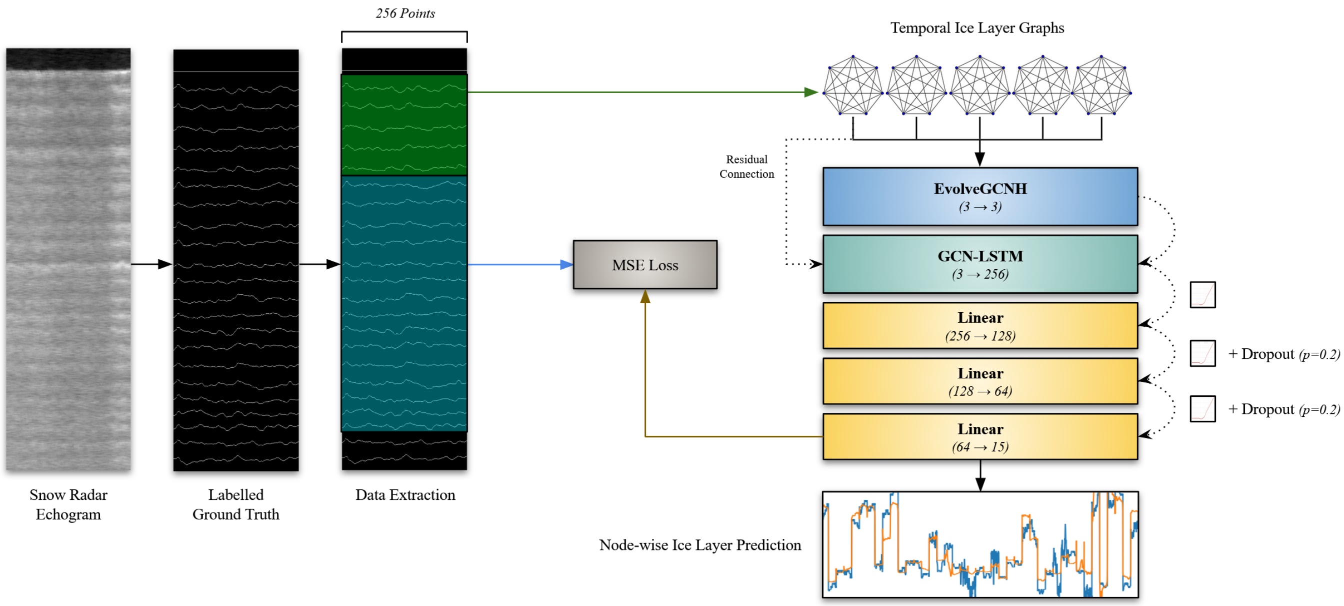

Fig. 1. Architecture of the proposed model.

structure of a graph. The size of the neighborhood on which convolution takes place is dictated by the number of sequential GCN layers present in the model (i.e. K GCN layers results in K-hop convolution). In a sense, GCNs are a generalized form of CNNs that enable variable degree.

A special form of GCN, known as adaptive GCN (or AGCN), define edge weights within an input graph as learnable parameters rather than predefined constants. In certain cases, this may increase model performance if relationships between nodes are more advanced than those specified by the input. In the case of our model, we route the graphs through an EvolveGCNH layer [9] prior to entering the GCN-LSTM layer.

EvolveGCNH is a version of EvolveGCN that behaves similarly to a traditional GCN, but treats its learned weight matrix as a temporal hidden state that, through use of a gated recurrent unit (GRU), implicitly adjusts the structure of input graphs by modifying node embeddings. The adjustment of the weight matrix at each forward pass is influenced by the previous hidden weight state as well as the node embeddings of the current input graph.

B. Physical Ice Features

In this paper, we experiment with five different physical properties as node features: snow mass balance, surface temperature, meltwater refreezing, height change due to melt, and snowpack height that are extracted from MAR model. "Snow mass balance" is the sum of daily snowfall minus sublimation, evaporation, and run-off. "Surface temperature" is the average annual surface temperature at that point during the respective layer's surface year. "Meltwater refreezing" is the annual amount of meltwater that refroze at that location. "Height change due to melt" is the annual change in height that an ice layer experienced due to melt. "Snowpack height" is the total height of snowpack (non-melting, dense snow) within a particular ice layer. We also experimented with using the elevation of the ice sheet as each point as a node feature.

C. Model Architecture

Our model (see Figure 1) uses an EvolveGCNH layer to introduce adaptivity to input adjacency matrices. The resulting node matrix is used as the feature matrix for a GCN-LSTM layer with 256 output channels. This leads into three fully-connected layers: the first with 128 output channels, the second with 64 output channels, and the third with 15 output channels, each corresponding to one of the 15 predicted ice layer thicknesses. Between each layer is the Hardswish activation function [21], an optimized approximation of the Swish function that has been shown to perform better than ReLU and its derivatives in deep networks [22]. Between the fully-connected layers is Dropout [23] with p=0.2. We use the Adam optimizer [24] over 300 epochs with mean-squared error loss. We use a dynamic learning rate that halves every 75 epochs beginning at 0.01.

D. Graph Generation

Each ground-truth echogram image is converted into five graphs, each consisting of 256 nodes. Each graph corresponds to a single ice layer for each year from 2007 to 2011. Each node represents a vertical column of pixels in the ground-truth echogram image and has three base features: two for the latitude and longitude at that point, and one for the thickness of the corresponding year's ice layer at that point. As part of the feature ablation study, we tested the model with every combination of five additional node features: snow mass balance, temperature, refreezing, height change, and snowpack.

TABLE I
RESULTS FROM THE NON-GEOMETRIC LSTM, NON-TEMPORAL GCN, NON-ADAPTIVE GCN-LSTM, NON-PHYSICAL AGCN-LSTM, AND BEST PROPOSED MODELS ON THE FIFTEEN PREDICTED ANNUAL ICE LAYER THICKNESSES FROM 1992 TO 2006. RESULTS ARE SHOWN AS THE MEAN $\pm$ STANDARD DEVIATION OF THE RMSE OVER FIVE TRIALS (IN PIXELS).

	LSTM	GCN	GCN-LSTM	AGCN-LSTM	Physical AGCN-LSTM
Total RMSE	5.817 ± 1.349	3.496 ± 0.509	2.766 ± 0.312	2.712 ± 0.179	$\mathbf{2.599 \pm 0.086}$

All graphs are fully connected and undirected. All edges are inversely weighted by the geographic distance between node locations using the haversine formula. For a weighted adjacency matrix A:

$$A_{i,j} = \frac{1}{2\arcsin\left(\mathrm{hav}(\phi_j - \phi_i) + \cos(\phi_i)\cos(\phi_j)\ \mathrm{hav}(\lambda_j - \lambda_i)\right)}$$

where

$$\mathrm{hav}(\theta) = \sin^2\left(\frac{\theta}{2}\right)$$

$A_{i,j}$ represents the weight of the edge between nodes i and j. ϕ and λ represent the latitude and longitude features of a node, respectively. Node features of all graphs are collectively normalized using z-score normalization. Weights in the adjacency matrices of all graphs are collectively normalized using min-max normalization with a slight offset to prevent zero- and one-weight edges. Self-loops are added with a weight of two. While we use an EvolveGCNH layer to introduce learned adjacency, this predefined spatial adjacency matrix serves as the initial state of the learned adjacency matrix, and is also passed residually to the GCN-LSTM layer.

V. RESULTS

In order to determine which physical parameters were important and which were not, we performed a node feature ablation study in which we tested how every possible combination of features performed with identical hyperparameters, input data, and random seeds. We found that using the combination of snow mass balance, meltwater refreezing, and height change due to melt as node features (in addition to latitude, longitude, and layer height) resulted in noticeably increased performance and consistency. This difference is highlighted in Table I, where the non-physical model (labelled "AGCN-LSTM") performs worse by an average of 0.11 pixels. The performance of other baseline models, including a non-geometric LSTM, non-temporal GCN, and non-adaptive GCN-LSTM are shown as well.

Over each trial, the root mean squared error (RMSE) was taken between the predicted and ground truth thickness values for each of the fifteen ice layers from 1992 to 2006 over all images in its corresponding testing set. The mean and standard deviation RMSE over all five trials are displayed in Table I. The proposed Physical AGCN-LSTM model consistently performed better than the baseline models in terms of mean RMSE. The qualitative results are depicted in Figure 2.

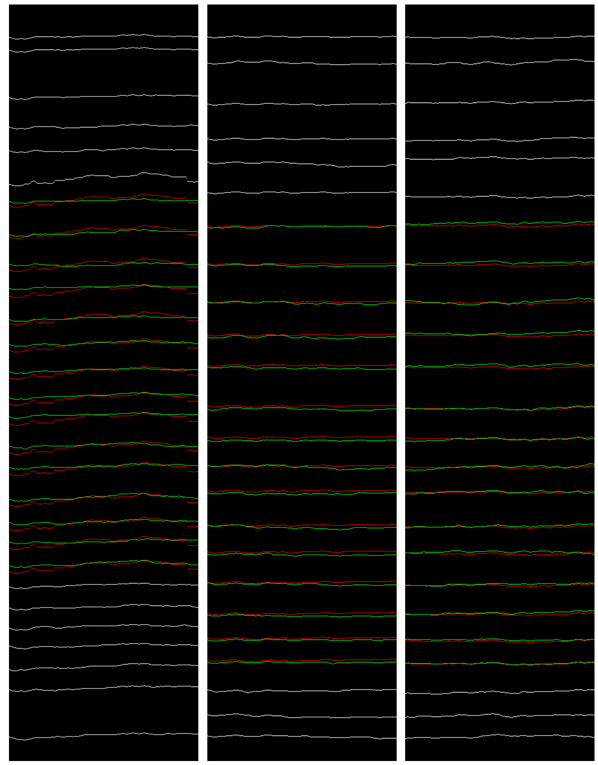

Fig. 2. Examples of model outputs. Green pixels represent ground truths, and red pixels represent predicted values.

VI. CONCLUSION

In this work, we developed a temporal, geometric, adaptive multi-target machine learning model that predicts the thicknesses of deep ice layers within the Greenland ice sheet (corresponding to the annual snow accumulation from 1992 to 2006, respectively) given the thicknesses of shallow ice layers (corresponding to the annual snow accumulation from 2007 to 2011, respectively) to consider physical parameters of the shallow ice layers during prediction. Our proposed model was shown to perform better and with more consistency than our previous, non-physical and baseline models.

ACKNOWLEDGMENT

This work is supported by NSF BIGDATA awards (IIS-1838230, IIS-1838024), IBM, and Amazon. We acknowledge data and data products from CReSIS generated with support from the University of Kansas and NASA Operation Ice-Bridge.

REFERENCES

[1] S. Gogineni, J. B. Yan, D. Gomez, F. Rodriguez-Morales, J. Paden, and C. Leuschen, "Ultra-wideband radars for remote sensing of snow and ice," in *IEEE MTT-S International Microwave and RF Conference*, 2013, pp. 1–4.

[2] Thomas N. Kipf and Max Welling, "Semi-supervised classification with graph convolutional networks," in *Proceedings of ICLR 2017*, 2017.

[3] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," 2017.

[4] Bing Yu, Haoteng Yin, and Zhanxing Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 2018.

[5] Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Jul. 2019, vol. 33, pp. 922–929.

[6] Dogan Aykas and Siamak Mehrkanoon, "Multistream graph attention networks for wind speed forecasting," *CoRR*, vol. abs/2108.07063, 2021.

[7] Damian Owerko, Fernando Gama, and Alejandro Ribeiro, "Predicting power outages using graph neural networks," 11 2018, pp. 743–747.

[8] Youngjoo Seo, Michaël Defferrard, Pierre Vandergheynst, and Xavier Bresson, "Structured sequence modeling with graph convolutional recurrent networks," in *Proceedings of ICLR 2017*, 2017.

[9] Aldo Pareja, Giacomo Domeniconi, Jie Chen, Tengfei Ma, Toyotaro Suzumura, Hiroki Kanezashi, Tim Kaler, Tao B. Schardl, and Charles E. Leiserson, "Evolvegcn: Evolving graph convolutional networks for dynamic graphs," 2019.

[10] Maryam Rahnemoonfar, Masoud Yari, John Paden, Lora Koenig, and Oluwanisola Ibikunle, "Deep Multi-Scale Learning for Automatic Tracking of Internal Layers of Ice in Radar Data," *Journal of Glaciology*, vol. 67, no. 261, pp. 39–48, 2021.

[11] Debvrat Varshney, Maryam Rahnemoonfar, Masoud Yari, and John Paden, "Deep Ice layer Tracking and Thickness Estimation using Fully Convolutional Networks," in *2020 IEEE International Conference on Big Data (Big Data)*. IEEE, 2020, pp. 3943–3952.

[12] Masoud Yari, Maryam Rahnemoonfar, John Paden, Ibikunle Oluwanisola, Lora Koenig, and Lynn Montgomery, "Smart Tracking of Internal Layers of Ice in Radar Data via Multi-Scale Learning," in *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, 2019, pp. 5462–5468.

[13] Masoud Yari, Maryam Rahnemoonfar, and John Paden, "Multi-Scale and Temporal Transfer Learning for Automatic Tracking of Internal Ice Layers," in *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, 2020, pp. 6934–6937.

[14] Danfei Xu, Yuke Zhu, Christopher B. Choy, and Li Fei-Fei, "Scene graph generation by iterative message passing," 2017.

[15] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh, "Graph r-cnn for scene graph generation," 2018.

[16] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon, "Dynamic graph cnn for learning on point clouds," 2018.

[17] Loic Landrieu and Martin Simonovsky, "Large-scale point cloud semantic segmentation with superpoint graphs," 2017.

[18] Tyler Wilson, Pang-Ning Tan, and Lifeng Luo, "A low rank weighted graph convolutional approach to weather prediction," in *2018 IEEE International Conference on Data Mining (ICDM)*, 2018, pp. 627–636.

[19] Benjamin Zalatan and Maryam Rahnemoonfar, "Recurrent graph convolutional networks for spatiotemporal prediction of snow accumulation using airborne radar," 2023.

[20] Benjamin Zalatan and Maryam Rahnemoonfar, "Prediction of deep ice layer thickness using adaptive recurrent graph neural networks," in *IEEE International Conference on Image Processing, ICIP 2023, Kuala Lumpur, Malaysia, October 8-11, 2023*. 2023, pp. 2835–2839, IEEE.

[21] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, Quoc V. Le, and Hartwig Adam, "Searching for mobilenetv3," *CoRR*, vol. abs/1905.02244, 2019.

[22] Prajit Ramachandran, Barret Zoph, and Quoc V. Le, "Searching for activation functions," 2017.

[23] Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," 2012.

[24] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," 2014.