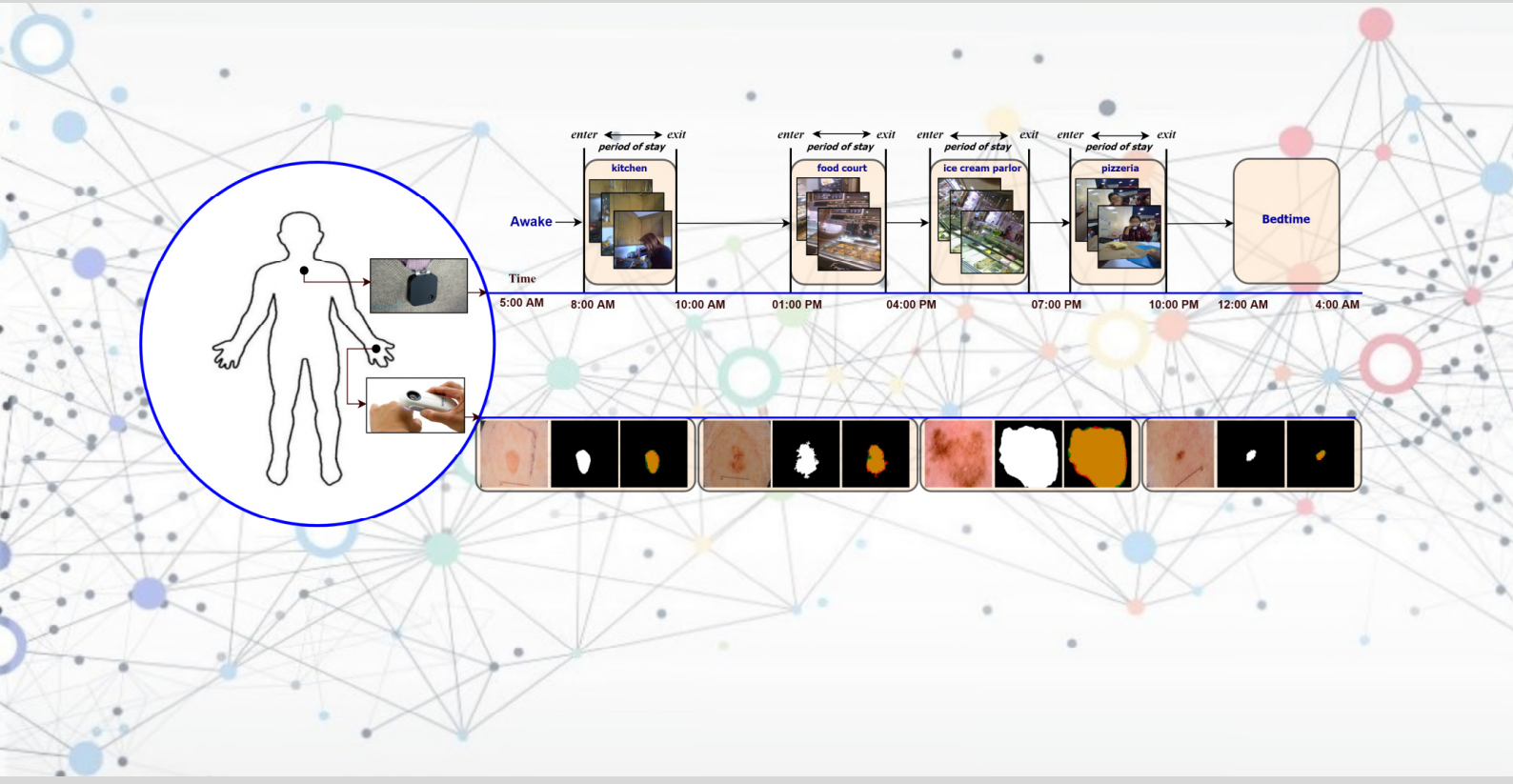
EFFICIENT DEEP LEARNING MODELS AND THEIR APPLICATIONS TO HEALTH INFORMATICS

Md Mostafa Kamal Sarker

UNIVERSITAT
ROVIRA i VIRGILI

Efficient Deep Learning Models and Their Applications to Health Informatics

MD MOSTAFA KAMAL SARKER



DOCTORAL THESIS
2019

UNIVERSITAT ROVIRA I VIRGILI
EFFICIENT DEEP LEARNING MODELS AND THEIR APPLICATIONS TO HEALTH INFORMATICS
Md Mostafa Kamal Sarker

Efficient Deep Learning Models and Their Applications to Health Informatics

DOCTORAL THESIS

Author:
Md. Mostafa Kamal Sarker

Advisors:
Dr. Domènec Savi Puig Valls
Dr. Petia Radeva

Departament d'Enginyeria Informàtica i Matemàtiques

UNIVERSITAT ROVIRA I VIRGILI
Tarragona
2019

UNIVERSITAT ROVIRA I VIRGILI
EFFICIENT DEEP LEARNING MODELS AND THEIR APPLICATIONS TO HEALTH INFORMATICS
Md Mostafa Kamal Sarker

UNIVERSITAT
ROVIRA I VIRGILI
Departament d'Enginyeria Informàtica
i Matemàtiques

Av. Paisos Catalans, 27
43007 Tarragona
Tel. +34 977 55 95 95
Fax. +34 977 55 95 97

We STATE that the present study, entitled "Efficient Deep Learning Models and their Applications to Health Informatics", presented by Md. Mostafa Kamal Sarker, for the award of the degree of Doctor, has been carried out under our supervision at the Departament d'Enginyeria Informàtica i Matemàtiques.

Tarragona, 12th September 2019.

Doctoral Thesis Supervisors,

Dr. Domènec Savi Puig Valls

Dr. Petia Radeva

UNIVERSITAT ROVIRA I VIRGILI
EFFICIENT DEEP LEARNING MODELS AND THEIR APPLICATIONS TO HEALTH INFORMATICS
Md Mostafa Kamal Sarker

To my son Mahrus Sarker, my wife Syeda Furruka Banu,
my mother Rizia Aziz, my father Md. Abdul Aziz Sarker,
my brothers and sisters.

UNIVERSITAT ROVIRA I VIRGILI
EFFICIENT DEEP LEARNING MODELS AND THEIR APPLICATIONS TO HEALTH INFORMATICS
Md Mostafa Kamal Sarker

Abstract

In this thesis, efficient deep learning methods are designed and implemented to solve classification and segmentation problems in two major areas of health informatics domains, namely pervasive sensing and medical imaging. In the area of pervasive sensing, this thesis focuses only on food and related scene classification for health and nutrition analysis. Recent studies show that it is important to know: *where we eat?* and *what we eat?* for properly monitoring our health conditions. To address those issues, deep learning models are employed by classifying food and related places. Moreover, the entire environment (e.g. create new datasets, models selection, parameter optimization, etc.) is prepared for this research. To handle the first issue, *where we eat?*, a new dataset is developed, named "FoodPlaces", which consists of 35 food-related places form different public datasets. Later, different state-of-the-art convolutional neural network (CNN) models are evaluated on this dataset by fine-tuning their parameters using transfer learning. Inspired by the outcomes of the first analysis, another dataset is developed, named "EgoFoodPlaces", using the wearable camera with 22 food places where a user of the camera, called "first-person", often visited. Afterwards, a new architecture based on multi-scale atrous convolutional networks is designed, named "MACNet", for evaluating image-level classification on this dataset. An overall comparable accuracy is achieved in this experiment for all classes of this dataset, where each class refers to a different food place, such as bar, coffee shop and restaurant, etc. In order to study the temporal information and correlation between the frames captured by the egocentric camera, the problem is redefined based on the appropriate temporal intervals (period of stay). This period is then split into a set of events which is a sequence of correlated frames. Thus, a novel attention-based deep network, named "MACNet+SA", is introduced using previously defined "MACNet" model with self-attention mechanism for improving the classification rate of food places. The model "MACNet+SA" has set a state-of-the-art classification result using event-level analysis of egocentric photo-streams using "EgoFoodPlaces" dataset. To deal with

the second issue, *what we eat?*, another new dataset is developed with food attributes, called "Yummly48K", which aims to analyze food nutrition by classifying cuisine and food flavour. Eventually, a multi-scale convolutional network is presented, named "CuisineNet", which is designed by aggregating convolution layers with various kernel sizes followed by residual and pyramid pooling module with two fully connected pathway. This model is introduced to solve the multi-modal classification problems for cuisine and flavours.

In the field of medical imaging, this thesis targets skin lesion segmentation problem in the dermoscopic images. In this research, two novel deep learning models are introduced to accurately segment the skin lesions. Firstly, a robust deep learning model is designed as an encoder-decoder network, called "SLSDeep". The encoder network in "SLSDeep" is composed of dilated residual layers, in turn, a pyramid pooling network followed by three convolution layers is used for the decoder. Moreover, a new loss function is formed by fusing both Negative Log-Likelihood (NLL) and End Point Error (EPE) to accurately segment the melanoma regions. Secondly, a lightweight and efficient model based on Generative Adversarial Networks (GANs), called "MobileGAN", is proposed for skin lesion segmentation. The "MobileGAN" combines 1D non-bottleneck factorization networks with position and channel attention modules in a conditional Generative Adversarial Networks (cGANs) model. The proposed model has only a few (2.35 million) parameters and is faster than the other state-of-the-art models. The International Symposium on Biomedical Imaging (ISBI) 2016, 2017 and International Skin Imaging Collaboration (ISIC) 2018 benchmark datasets are used for the skin lesion segmentation task for evaluating the proposed models. Both proposed models present comparable and better segmentation accuracy than the state-of-the-art skin lesion segmentation models.

Keywords: Deep Learning, Wearable Device, Food Places Classification, Convolutional Neural Network, Recurrent Neural Network, Skin Lesion Segmentation, Dilated Convolutional Neural Network, Generative Adversarial Network.

Acknowledgements

‘Al-hamdu lillahi Rabbil-‘alamin, All the praises and thanks be to Allah, the Rubb of ‘Alamin (the Supreme Lord of mankind and all that exists) for his kindness and blessings upon us on the occasion of the successful completion of this thesis.

First and foremost, I would like to thank my supervisors, Professor Domenec Puig and Professor Petia Radeva for their invaluable guidance, support, motivation, and encouragement. Thank you Professor Domenec for believing in me and allowing me the opportunity to realize the PhD studies. Thank you Professor Petia for giving me the best suggestions and precious guidance. My sincere gratitude also goes to Dr. Sylvie Chambon, for providing me with the opportunity of joining their team for mobility visit in the INP-ENSEEIHT, Toulouse, France, for her passionate interest, and precious guidance. Thank you, Sylvie.

My greatest appreciation goes to my wife, Syeda Furruka Banu, my parents, for the constant input and prayer. I truly want to name every friend that I made in the IRCV lab. To Adel, Vivek, for all the funny fights, chats, laughter, and foods. Also for the helping and keeping me company in the empty URV during the sleepless nights. To Hatem A. Raswan for his continuous help and support. To Mohamed Abdel-Nasser for his most sincere friendship, support and famous jokes, which refreshing me for doing lots of hard works. To Farhan for pure kindness and lovely friendship and support. To Julian, for his invaluable support over these years. I also would like to thanks Saddam Abdulwahab, Mohammed Jabreel, Emran, Fadi, Abdulrahman for all the memories, experiences, travels, and funs. Thank you guys, without you IRCV is not any special place. I am also grateful for the support granted by the program "Marti Franques" under the agreement between Universitat Rovira Virgili and Fundació Catalunya La Pedrera.

Last, but not the least, my most sincere appreciation to Nazmul Vai and Saikat for making the last miles of this marathon the loveliest to me. Thank you. Gracias.

List of Publications

1. **Md. Mostafa Kamal Sarker**, Hatem A. Rashwan, Farhan Akram, Estefania Talavera, Syeda Furruka Banu, Petia Radeva, and Domenec Puig, *"Recognizing Food Places in Egocentric Photo-streams using Multi-scale Atrous Convolutional Networks and Self-Attention Mechanism"*. IEEE Access, pp. 39069 – 39082, Vol. 7, 2019.(**Q1, Impact factor 2019: 4.09**)
2. Estefania Talavera, Maria Leyva-Vallina, **Md. Mostafa Kamal Sarker**, Domenec Puig, Nicolai Petkov and Petia Radeva, *"Hierarchical approach to classifying food scenes in egocentric photo-streams"*. in IEEE Journal of Biomedical and Health Informatics. 2019. (**Q1, Impact factor 2019: 4.217**)
3. **Md. Mostafa Kamal Sarker**, Hatem A. Rashwan, Estefania Talavera, S. Furruka Banu, Petia Radeva, and Domenec Puig, *"MACNet: Multi-scale Atrous Convolution Networks for Food Places Classification in Egocentric Photo-Streams"*. In: European Conference on Computer Vision - ECCV 2018 (EPIC@ECCV WS). September 8 -14, 2018, Munich, Germany. (**CORE ranking: A**)
4. **Md. Mostafa Kamal Sarker**, Hatem A. Rashwan, Syeda Furruka Banu, Adel Saleh, Vivek Kumar Singh, Forhad Chowdhury,*"SLSDeep: Skin Lesion Segmentation Based on Dilated Residual and Pyramid Pooling Networks"*. In: Medical Image Computing and Computer Assisted Intervention – MICCAI 2018. September 16-20 2018, Granada, Spain. (**CORE ranking: A**)
5. Singh, Vivek Kumar, Santiago Romani, Hatem A. Rashwan, Farhan Akram, **Md. Mostafa Kamal Sarker**,*"Conditional Generative Adversarial and Convolutional Networks for X-ray Breast Mass Segmentation and Shape Classification"*. In: Medical Image Computing and Computer Assisted Intervention – MICCAI 2018. September 16-20 2018, Granada, Spain. (**CORE ranking: A**)

6. Singh, Vivek Kumar, Hatem Rashwan, Farhan Akram, Nidhi Pandey, **Md. Mostafa Kamal Sarker**, Adel Saleh et al. *"Breast Tumor Segmentation and Shape Classification in Mammograms using Generative Adversarial and Convolutional Neural Network"*. Expert Systems with Applications. pp.112855, July 2019. (**Q1, Impact factor 2019: 4.292**)
7. **Md. Mostafa Kamal Sarker**, Mohammed Jabreel, Hatem A. Rashwan, Syeda Furruka Banu, Petia Radeva, and Domenec Puig. *"CuisineNet: Food Attributes Classification using Multi-scale Convolution Network"*. In 21th International Conference of the Catalan Association for Artificial Intelligence (CCIA 2018), 8-10th October 2018, Roses, Spain.
8. Singh, Vivek Kumar, Hatem Rashwan, Farhan Akram, Nidhi Pandey, **Md. Mostafa Kamal Sarker**, Adel Saleh et al.*"Retinal Optic Disc Segmentation using Conditional Generative Adversarial Network"*. In 21th International Conference of the Catalan Association for Artificial Intelligence (CCIA 2018), 8-10th October 2018, Roses, Spain.
9. Farhan Akram, Miguel Angel Garcia, Vivek Kumar Singh, **Md. Mostafa Kamal Sarker** and Domenec Puig, *"Brain MR Image Segmentation Using Multiphase Active Contours Based on Local and Global Fitted Images"*. In 21th International Conference of the Catalan Association for Artificial Intelligence (CCIA 2018), 8-10th October 2018, Roses, Spain.
10. Adel Saleh, Mohamed Abdel-Nasser, **Md. Mostafa Kamal Sarker**, Vivek Kumar Singh, Saddam Abdulwahab, Nasibeh Saffari, Miguel Angel Garcia, and Domenec Puig. *"Deep visual embedding for image classification"*. In 2018 International Conference on Innovative Trends in Computer Engineering(ITCE), 19t-21 February 2018, Aswan, Egypt.
11. **Md. Mostafa Kamal Sarker**, Maria Leyva, Adel Saleh, Vivek Kumar Singh, Farhan Akram, Petia Radeva and Domenec Puig. *"FoodPlaces: Learning Deep Features for Food Related Scene Understanding"*. In 20th International Conference of the Catalan Association for Artificial Intelligence (CCIA 2017). 25-27th October 2017, Deltebre, Spain.

12. Vivek Kumar Singh, Santiago Romani, Jordina Torrents-Barrena, Farhan Akram, Nidhi Pandey, **Md. Mostafa Kamal Sarker**, Adel Saleh, Meritxell Arenas, Miguel Arquez and Domenec Puig, *"Classification of Breast Cancer Molecular Subtypes from their Micro-Texture in Mammograms using a VGGNet-Based Convolutional Neural Network"*. In 20th International Conference of the Catalan Association for Artificial Intelligence (CCIA 2017). 25-27th October 2017, Deltebre, Spain.
13. Farhan Akram, Miguel Angel Garcia, Vivek Kumar Singh, **Md. Mostafa Kamal Sarker** and Domenec Puig, *"Image segmentation using active contours driven by bias fitted image robust to intensity inhomogeneity"*. In 20th International Conference of the Catalan Association for Artificial Intelligence (CCIA 2017). 25-27th October 2017, Deltebre, Spain.
14. **Md. Mostafa Kamal Sarker**, Hatem A. Rashwan, Mohamed Abdel-Nasser, Vivek Kumar Singh, Syeda Furruka Banu, Farhan Akram, Forhad U H Chowdhury, Kabir Ahmed Choudhury, Sylvie Chambon, Petia Radeva, Domenec Puig. *"MobileGAN: Skin Lesion Segmentation Using a Lightweight Generative Adversarial Network"*. arXiv preprint arXiv:1907.00856. (submitted to IEEE Transactions on Biomedical Engineering).
15. Vivek Kumar Singh, Hatem A. Rashwan, Mohamed Abdel-Nasser, **Md. Mostafa Kamal Sarker**, Farhan Akram, Domenec Puig. *"An Efficient Solution for Breast Tumor Segmentation and Classification in Ultrasound Images Using Deep Adversarial Learning"*. arXiv preprint arXiv:1907.00887.
16. Farhan Akram, Vivek Kumar Singh, Hatem A Rashwan, Mohamed Abdel-Nasser, Sarker, **Md. Mostafa Kamal Sarker**, Domenec Puig,*"Adversarial Learning with Multiscale Features and Kernel Factorization for Retinal Blood Vessel Segmentation"*. arXiv preprint arXiv:1907.02742.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Today, Artificial Intelligence (AI) is a flourishing area with lots of practical applications and active research topics. The future application of AI is expanding dramatically, including more self-driving cars, intelligent robots, healthcare diagnostics, precision medicine, and others. Recently, the progress in information and communication technology have made a remarkable revolution in various disciplines, including medicine and public health. Due to these advances, there is an enormous amount of data generated daily from individuals. Processing and obtaining important information from collected data in real-life activities is a challenging task. The performance of the data analytics in health informatics has grown promptly with this large influx of multi-modality data in the last decade. Health informatics illustrates the use of healthcare information related data to boost patient care across interactions with the health system. Nowadays, Deep Learning (DL), an advanced form of traditional Machine Learning (ML), allowed developing computational models that are made of different processing layers based on neural networks to learn representations of data with various levels of abstraction. In fact, DL provides state-of-the-art achievement in various fields of AI, in particular for image or scene classification, object detection, semantic segmentation and so on in computer vision area. Several Deep Neural Networks (DNNs) models (e.g., especially,

Convolutional Neural Networks (CNNs) based) such as AlexNet (Krizhevsky et al., 2012), VGGNet (Simonyan and Zisserman, 2014), GoogleNet (Szegedy et al., 2016), ResNet (He et al., 2016) have successfully performed state-of-the-art performance in the domain of image classification, detection and segmentation. In classification, the goal is to calculate the class probability of a given example referring to each output class where the output is a vector of class confidence. The popular models for the image classification tasks are including AlexNet (Krizhevsky et al., 2012), VGGNet (Simonyan and Zisserman, 2014), Inception-V3 (Szegedy et al., 2016), Inception-ResNet-V2 (Szegedy et al., 2017), ResNet (He et al., 2016), DenseNet (Huang et al., 2017) and so on.

For segmentation, traditionally DL-based semantic segmentation models are used for the segmentation tasks, such as FCN (Long et al., 2015), SegNet (Badrinarayanan et al., 2017), U-Net (Ronneberger et al., 2015), RefineNet (Lin et al., 2017b), DeepLab (Chen et al., 2016a), etc. However, all these DL models are performing better against traditional ML for many research outcomes, such as recently proposed activation function, Rectified Linear Unit (ReLU) (Nair and Hinton, 2010) which resolve the training problem, dropout helps to regularize the networks for efficient training and various effective optimization methods for training DL models efficiently. Moreover, in many cases, the DL models are required large scale datasets for training and evaluating in different problems, such as ImageNet (Russakovsky et al., 2015a), and MSCOCO (Lin et al., 2014) provides the state-of-the-art performance of DL models for image classification and semantic segmentation. The main advantages of DL models are that one model can be employed in different domains in health informatics (such as pervasive sensing and medical imaging). The success of CNNs in the field of computer vision inspired to apply DL approaches in different modalities in pervasive sensing, such as food and related place classification and human activity recognition from images or frames using wearable or mobile camera, also in medical imaging including segmentation, classification, detection, registration, and medical information processing. Meanwhile, these DL models performed tremendous achievement in the different modalities of pervasive sensing, such as a mobile-based

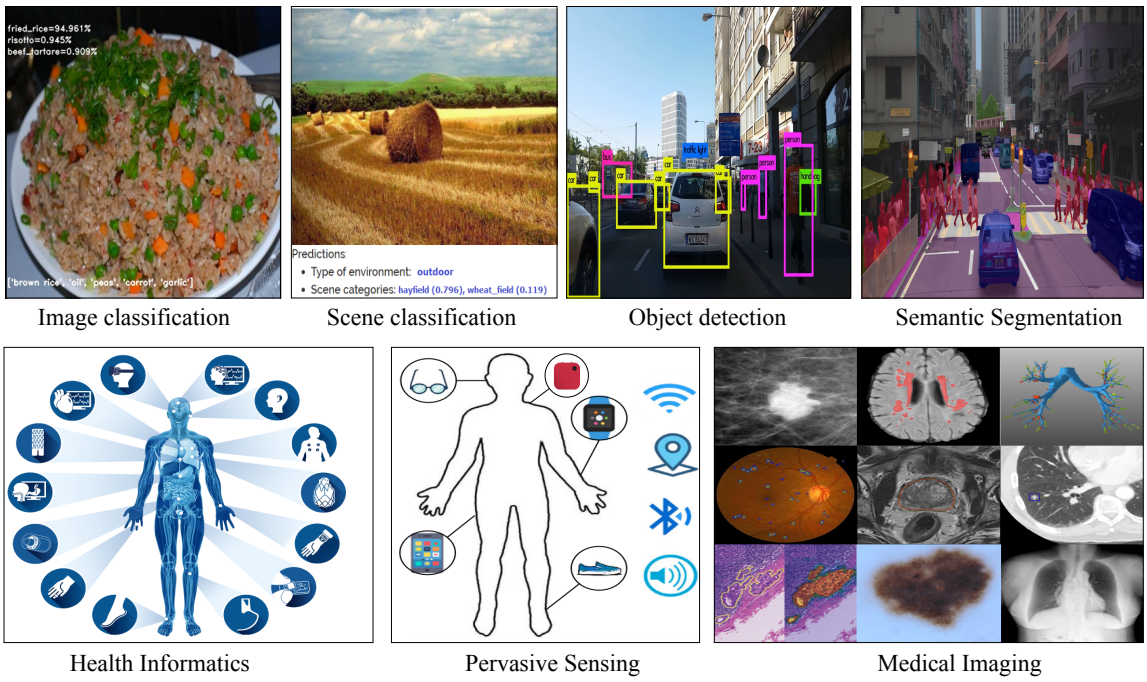


Figure 1.1: Some examples of the successful DL application domains where DL achieved state-of-the-art performance.

diet monitoring system (e Silva et al., 2018) and activity monitoring for recognition of older people's d daily routine in order to improve their lifestyle (Wang et al., 2018), in medical imaging, Dermatologist-level performance for skin cancer detection (Esteva et al., 2017), breast cancer segmentation, classification and detection (Singh et al., 2019), lung cancer detection, Neuroimaging for analysis Brain Tumor (Akkus et al., 2017) and so on. Some examples applications of DL models are shown in Figure 1.1.

Currently, overweight and obesity are major health problems in high-income countries. The yearly health-care cost of obesity in the US was $147 billion in 2008 and the medical cost for people who have obesity was $1,429 higher than those of normal weight (Finkelstein et al., 2009). The obesity medical cost (direct and indirect) in Europe was estimated at around €81 billion per year in 2012. In keeping with the WHO estimates on obesity expenditure, this was 2%–8% of the total national expenditure in the 53 European countries (Cuschieri and Mamo, 2016). Moreover, obesity has already been verified as a risk factor for several cancers, including endometrial, liver, kidney, colorectal, and pancreatic cancer. The most obvious risk factor for skin cancer is unprotected sun exposure. However, according

to earlier studies, obesity may also play a role. Recently, researchers set out to further investigate obesity's role in the risk of melanoma, a quick-growing form of skin cancer (Newman, 2018). According to Signify Research (Research, 2017), it is expected that applications of deep learning for medical imaging alone will be funded more than $300 million by 2021, which is higher than the cost of the whole analysis industry spent in 2016 (Han et al., 2017). A good example of it is given by IBM Watson - a rule-based expert system for medical diagnosis developed by the radiology application of Dr. Watson. Recently, Google also invested billions of the dollar for their medical imaging project, "Deep Mind Health", another giant in this domain. Global AI based on DL approaches in healthcare spent almost about $8.6 billion in 2017. Finally, the investment is growing on and on for developing DL based system in every application domain.

However, the main motivation of this thesis is to contribute for developing efficient DL-based system to be able to prevent obesity and early diagnosis of skin cancer. The system will help doctors and dermatologists for making a more suitable decision which finally will secure more careful treatment of the patients.

1.2 Deep Learning

Machine Learning, a subset of AI, has transformed many fields into a new era for several decades. Artificial neural networks (ANN) is a sub-field of ML which leads to spawned DL. From the beginning, DL has been performing consistently with outstanding success in mostly every application domain. The taxonomy of AI is illustrated in Figure 1.2. Since 2006, employing either deep network architecture of learning or hierarchical learning methods were developed widely onward and become very popular with the name "Deep Learning". Learning is a method consisting of calculating the model parameters (weights) so that the learned model can complete a particular task. For instance, the parameters of ANN are associated with a weight that dictates the significance of the relationship in the neuron when multiplied by the input value, called weight matrices. On the other hand, DL

consists of different layers between the input and output layer which enables many steps of non-linear information processing units by hierarchical architectures to be present that are utilized for feature learning and pattern classification. The procedure of learning based on the representations of data is defined as representation learning (Schmidhuber, 2015). Recent research outcome states that the DL-based representation learning includes a hierarchy of features, where high-level features are perhaps determined from the low-level ones and vice versa. Most of the literature is explained about DL as a universal learning method and can solve nearly all types of problems in the different application field. More specifically, DL is not for solving a particular task only. Currently, DL is being applied in almost every domain and performs outstanding outcomes which changed the era of AI.

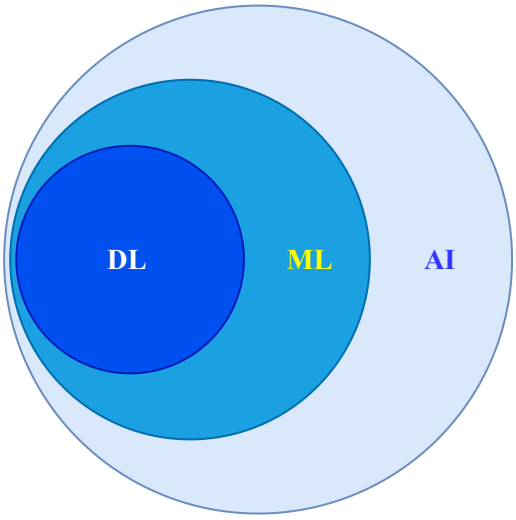


Figure 1.2: The taxonomy of AI.

DL models are categorized as follows: supervised, semi-supervised and unsupervised. ***Supervised Learning*** is a learning procedure that requires labeled data. In supervised DL, the context needs a set of inputs with its corresponding outputs. For example, if the input is x_t, the system predicts $\hat{y}_t = f(x_t)$ and it receives a loss value $l(y_t, \hat{y}_t)$, where y_t is target output and $\hat{y}_t$ is predicted output. The system then iteratively adjusts the network parameters for better estimation of the target outputs. The system can get the correct answer to questions from the context. Several deep supervised learning models are

available including DNNs, CNNs, and Recurrent Neural Networks (RNNs) (e.g., including Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU)). ***Semi-supervised Learning*** is based on partially labeled data, often also called Reinforcement Learning (RL). Moreover, Deep Reinforcement Learning (DRL) and Generative Adversarial Networks (GANs) (Schmidhuber, 2015) are sometimes used as semi-supervised learning models. Furthermore, RNNs including LSTM and GRU are also utilized for semi-supervised learning. ***Unsupervised Learning*** is a learning system that can learn the representation of the data without the presence of its labels. More precisely, the system learns the internal representation or essential features to identify unknown associations or distribution within the input data. In an unsupervised learning system, clustering, dimensionality minimization, and generative methods are often used. Many deep learning models are good at clustering and non-linear dimensionality minimization, such as Auto-Encoders (AE), Restricted Boltzmann Machines (RBM), and also recent GANs (Schmidhuber, 2015). Besides, RNNs sometimes are additionally used for unsupervised learning in several application areas (Schmidhuber, 2015).

1.3 Advantages and properties of Deep Learning

- The DL approach is often called as *universal learning approach*, because it can be implemented to almost every application domain.
- The *robustness* of DL approaches does not require to design any features in advance. Features are automatically learned with the variations of data that is best for the task.
- The *generalization* of deep learning models are that the same models can be used in different applications with different types of data by using the transfer learning property. Transfer learning is the improvement of learning a new task through the transfer of knowledge from a related task that has already been learned (Olivas, 2009). Moreover, this method is beneficial to solve the problem while it has a handful of data.

- The DL method is extremely *scalable*. In 2017, Google presented a deep network named "Inception-V4" (Szegedy et al., 2017) with 779 layers which was implemented on a supercomputing scale. This article shows how deep learning can deal with a variety of big data and has several advantages. Deep learning is a data-driven approach. The performance between traditional ML and DL approach is illustrated in Figure 1.3 with the evaluation of data scale.

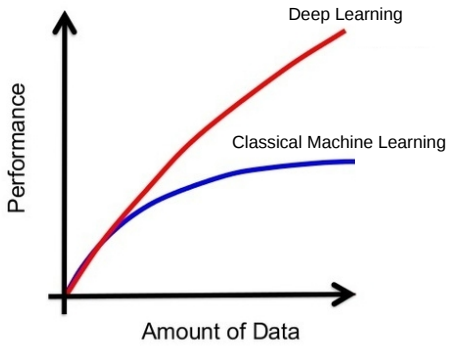


Figure 1.3: Performance of traditional ML and DL with respect to the amount of data.

1.4 Deep Learning in Health Informatics

In health informatics, the generation of automatic feature set without human intervention has many advantages. Therefore, DL methods (in particular, CNNs) have had the greatest impact within the field of health informatics. In this thesis, we applied our proposed efficient DL models for food and food-related places analysis in conventional and egocentric images and skin lesion segmentation that belongs to the field of pervasive sensing and medical imaging analysis.

1.4.1 Deep Learning in Pervasive Sensing

Pervasive sensors, such as implantable, wearable, and mobile devices (Yang et al., 2014) provide continuous monitoring of health and wellbeing. A proper calculation of food intake, food eating pattern, and energy expenditure during the day, for instance, can assist to prevent obesity and improve individual well-being. For elderly people

with chronic disorders, wearable devices have been used to enhance the quality of care by allowing people to continue living freely in their houses (Malwade et al., 2018). The supervision of patients with disabilities and undergoing rehabilitation can also be improved by analyzing patient's activity using wearable and implantable assistive sensors. For patients with crucial care, constant monitoring of vital symptoms, such as blood pressure, breath rate, and body temperature, are essential for advancing therapy outcomes by closely examining the patient's status (Ravì et al., 2016). Nowadays, physical trackers are becoming more popular in the pervasive sensor domain that record information such as the number of steps, heart rate, or social media messages. These lifelogging devices collect data (images, videos, sounds, speed, location, cardiac frequency, etc.), and investigate them, giving the user with information about their habits, such as steps walked, hours slept, social interactions, daily activity, etc. Particularly, wearable cameras, by capturing images frequently, provide visual information about the user's daily life; performing activities, participated events, visited places and social interactions of the first person. The wearable camera provides the capture of richer information from a first-person perspective of the user's daily life experiences. Through the study of the egocentric sequence of images (photo- streams), different methods have attempted to improve the first-person quality of life; investigating social interactions (Aghaei et al., 2018, 2015), important moments to be retrieved Bolanos et al. (2015) or building story-lines of first-person days (Bolanos et al., 2017, 2015). Finally, the motivation is driving towards the automatically analysis of people's daily health habits and food intake using the egocentric images captured by wearable camera (Bolaños and Radeva, 2016).

1.4.2 Deep Learning in Medical Imaging

Automatic medical imaging analysis is essential to modern medicine. Diagnosis based on the analysis of images can be extremely subjective where the structural abnormalities are recognized and classified into disease categories. Computer-aided diagnosis (CAD) can provide an accurate and faster estimation of the underlying

disease processes and ensure better treatment of a huge number of people simultaneously. The medical imaging appears from many imaging systems such as Microscopic, Histopathology, Computer Tomography (CT), ultrasound, Mammograms, X-ray, Magnetic Resonance Imaging (MRI), and Positron Emission Tomography (PET). There are numerous CAD systems developed and launched in the clinical workflow in early 2010. An efficient automatic CAD system without human engagement can help to reduce human errors, processing time and cost.

For the slow process of manual segmentation procedures in medical imaging, there is a significant interest in developing fast and accurate automatic segmentation algorithms that can segment accurately without human interaction. Yet, there are still some boundaries of medical image processing including data insufficiency and imbalanced classes. Usually, a large number of labelled data (commonly in thousands) for training is unavailable for many reasons (Schmidhuber, 2015). Labelling the sample needs domain specialist for that area which is costly and requires enough labour and time. However, to deal with this issue, many data transformation or augmentation methods (rotation, cropping, translation, whitening, and scaling) are used for increasing the amount of available labelled examples (Litjens et al., 2017; Greenspan et al., 2016). Sometimes, patch-based methods are applied to solve class imbalanced issues. In this thesis, we use data augmentation techniques to manage this problem. Several DL-based methods have been used in different areas in medical imaging including classification, segmentation, registration, computer-aided detection and diagnosis and precision imaging for personalized medicine. DL based methods have huge success in the medical image classification area. In computer vision area, ImageNet (that consists of 14 million samples with 1000 classes) carried out a great breakthrough for developing efficient DL models (Russakovsky et al., 2015b). Now, transfer learning is commonly applied using the DL models developed by training with ImageNet. The main two reasons to apply transfer learning are the use of pre-trained weights for feature extraction and use of pre-trained weights to a current network fine-tuned with a new dataset. For example, various researches have been conducted where the trained weights are adopted from ImageNet dataset and

many studies have described the best results for medical image classification with transfer learning (Litjens et al., 2017; Greenspan et al., 2016).

Outstanding work for skin cancer classification is used transfer learning with Google Inception-V3 model and achieved dermatologist-level performance (Esteva et al., 2017). DL methods are heavily used for neuroimaging of Brain Tumor and Alzheimer disease classification and segmentation based on 2D and 3D convolutional kernels approaches using MRI and achieved state-of-the-art performance (Zacharaki et al., 2009; Suk and Shen, 2016). Besides, DL techniques are also applied for the analysis of X-ray images from chest radiographs (Lakhani and Sundaram, 2017). Moreover, the success of DL approaches explored in different domain of medical imaging achieved state-of-the-art performance (Maier et al., 2019), i.e., Breast cancer detection and classification from histology images (Han et al., 2017), GAN-based Breast cancer segmentation and classification from mammogram (Singh et al., 2019), Double CNNs for Lung cancer segmentation and classification from CT images (Jakimovski and Davcev, 2019), Locality sensitive DL model Colon cancer segmentation and classification from histology images (Sirinukunwattana et al., 2016), Skin lesion segmentation based on GANs (Sarker et al., 2019a), classification-based on transfer learning (Esteva et al., 2017) and so on. Finally, the success story is adding more and more on the application of deep learning in medical imaging every day.

1.5 Research Contributions

This thesis demonstrates the design, development and application of deep learning models in two main areas (pervasive sensing and medical imaging) of the health informatics domain. In pervasive sensing, we mainly focused on food-related places classification for identifying food eating behaviour and intake from conventional and egocentric images using traditional and wearable cameras. On the other hand, in medical imaging, we address the skin lesion segmentation in dermoscopic images. The main contributions of this thesis are summarized in the following:

- We analyzed the performance of different state-of-the-art deep learning models to recognize the food-related scenes in conventional images (see chapter 3). In chapter 3, we present a new dataset called "FoodPlaces" with selecting 35 common food-related scenes from the three public datasets "Places365" (Zhou et al., 2016), "ImageNet" (Russakovsky et al., 2015b) and "SUN397" (Xiao et al., 2010). We use transfer learning (fine-tune) approach with the recent state-of-the-art convolutional neural network (CNNs) models (VGG-16 (Simonyan and Zisserman, 2014), ResNet-50 (He et al., 2016) and Inception-V3 (Szegedy et al., 2016)) in order to learn deep features from images for classifying the food-related scenes. The main results of Chapter 3 is published in a conference in 2017.
- We introduced new DNNs models for Food Places Classification in Egocentric Photo-streams (see Chapter 4). In this chapter, we present two models: one is based on image-level analysis of food-related scenes, and another is based on the event-level analysis. To develop these two models, we use multi-scale Atrous convolution networks, called "MACNet" (Sarker et al., 2018c) for improving image-level classification rate and MACNet with self-attention mechanism (MACNet+SA) for improving event-level classification rate of food places. In this chapter, we prove the efficiency of the multi-scale mechanism for improving the classification accuracy. Moreover, we present an ablation study over the achieved results by every image-level and event-level analysis and describe the robustness of the event-level analysis over the image-level analysis. The main results of the Chapter 4 are published in the European Conference on Computer Vision- ECCV 2018 and IEEE Access journal 2019.
- We design a new DNNs model for food attributes classification in conventional images (see Chapter 5). In this chapter, we address the problem of identifying the food culture of people around the world and its flavour by classifying two main food attributes, cuisine and flavour. To solve this problem, we designed a new multi-scale convolution network ("CuisineNet") by the aggregation of multi-scale convolution layers with different kernel size for extracting more

accurate features from input images and weighting the features results from different scales. We also introduce a new food attributes classification dataset, called "Yummly48K" and evaluated our model on this dataset. We show that the new architecture achieved best results than all other state-of-the-art methods on this dataset. The main outcome of this chapter is reported in an international conference (Sarker et al., 2018a).

- We proposed two different pipelines for solving the problem of skin lesion segmentation (see Chapter 6 and Chapter 7). In Chapter 6, we present a robust DNNs model based on dilated residual and pyramid pooling networks for skin lesion segmentation. We present a robust deep learning SLS model, called "SLSDeep", represented as an encoder-decoder network. The encoder network is constructed by dilated residual layers, in turn, a pyramid pooling network followed by three convolution layers is used for the decoder. we also provide an extensive comparison with the baseline models to highlight the significance of each component of our proposed model. In Chapter 7, we propose a lightweight and efficient GANs model, called "MobileGAN" for skin lesion segmentation. The MobileGAN combines 1D non-bottleneck factorization networks with position and channel attention modules in a GANs model. To prove the robustness and efficiency of our proposed models, a comparison with relevant state-of-the-art models are provided. The main results of Chapter 6 is published in MICCAI 2018 conference and Chapter 7 is submitted to the journal of IEEE Transactions on Biomedical Engineering.

1.6 Thesis Organization

This thesis aims to develop efficient deep learning models for food-related places classification and skin lesion segmentation. As we discussed, DNNs provides excellent performance in Section 1.1. Therefore, we also apply DNNs for solving the above two problems. The principal concept behind DNNs is to learn a representation that performs classes linearly separable. This thesis starts by presenting an insight

into the background DNNs in the Chapter 2. The thesis is mainly divided into two parts based on the application of proposed efficient DL models, for food and related environment analysis and skin lesion segmentation. For food and related environment analysis, we initially studied the food-related scene classification problem in conventional images for the first time and applied different popular DL models with our created dataset ("FoodPlace") which collected from differed publicly available datasets in Chapter 3. We design two DNNs models in Chapter 4 based on multi-scale atrous convolution networks ("MACNet") and self-attention mechanisms ("MACNet+SA") for image-level and event-level recognizing of food places in egocentric photo-streams and evaluate them using our own created dataset ("EgoFoodPlaces"). Our model provides more accurate classification accuracy than all other state-of-the-art DNNs models on this dataset. Chapter 5 introduces a novel multi-scale convolution network ("CuisineNet") for classification of food attributes in Conventional Images using our own dataset ("Yummly48K"). For medical image analysis, we employ the DNNs models for solving the issues of skin lesion segmentation problem in dermoscopic images. Firstly, we present a robust deep learning SLS model ("SLSDeep") represented as an encoder-decoder network based on Dilated Residual Networks (DRNs) and Pyramid Pooling Networks (PPNs) in Chapter 6. Later on, we present a novel lightweight and efficient GANs model, called "MobileGAN" for skin lesion segmentation and explain it in Chapter 7. The proposed model is computationally efficient and more accurate than many of the recent state-of-art models for skin lesion segmentation. Finally, the main conclusions and discussions about the possible future works are highlighted in Chapter 8.

Chapter 2

Efficient Deep Learning

2.1 Introduction

Inventors have long craved for creating machines that think (Goodfellow et al., 2016). Historically, these desire periods back to the era of ancient Greece, when the mythical figures Daedalus, Pygmalion and Hephaestus perhaps interpreted as legendary creators, and Talos, Galatea and Pandora may all be regarded as artificial life (Martin, 2004; Sparkes, 2013; Tandy and Neale, 1996). After the first invention of programmable computers by Charles Babbage (Menabrea and Lovelace, 1842), people are starting to think that such machines might become intelligent. In 1943, Walter Pitts, a logician, and Warren McCulloch, a neuroscientist, invented the first mathematical model of a neural network, called McCulloch-Pitts neurons (McCulloch and Pitts, 1943). They show that a combination of mathematics and algorithms can construct a system, which aimed to mimic human thought processes. Later in 1950, Turing proposed such a machine, even implying at genetic algorithms in his article "Computing Machinery and Intelligence" (Machinery, 1950). He designed what has been designated "The Turing Test"(The Imitation Game) to decide whether a computer can think. Afterwards, in 1952, Arthur Samuel creates the first learning programs for the computer (Samuel, 1962). The programs were constructed to play the checker game which based on correcting the mistakes and finding better ways to win from that data. This was the first examples of machine learning. Frank

Rosenblatt, a psychologist, shows that perceptrons will gather if what they are trying to learn can be represented and is broadly acknowledged as the foundation of Deep Neural Networks (DNNs) (Rosenblatt, 1957). In (Minsky and Papert, 1969), Minsky & Papert demonstrates the limitations of perceptron's that stopping research in neural networks for a decade. Geoffrey Hinton renovates the field by backpropagation algorithm (Ackley et al., 1985) in 1985. Eventually, a hierarchical neural network called Neocognitron (Fukushima, 1988), which capable of visual pattern recognition in 1988. The fruitful application of Convolutional Neural Networks (CNNs) with backpropagation for document analysis (LeCun et al., 1998) proposed by Yan LeCun in 1988. Later, Geoffrey Hinton lab resolves the training problem for DNNs (Hinton et al., 2006). The final breakthrough of DNNs comes after introducing AlexNet by Alex Krizhevesky (Krizhevsky et al., 2012), which won the first place in ImageNet: Large Scale Visual Recognition Competition (ILSVRC) competition, 2012.

The theoretical frameworks of DL are well rooted in the traditional neural networks (NNs) research. Yet different to the more conventional use of NNs, deep learning considers for the use of multiple hidden neurons and layers, usually more than two, as an architectural success combined with modern training paradigms. During resorting to multiple neurons enables an extensive coverage of the raw data at hand, the layer-by-layer pipeline of a nonlinear sequence of their outputs produces a lower-dimensional projection of the input space. Each lower-dimensional projection corresponds to a higher perceptual level. Provided that the network is optimally weighted, it outcomes in an efficient high-level abstraction of the raw data or images. This high level of abstraction provides an automatic feature set, which differently would have expected hand-crafted or bespoke features. Among numerous methodological alternatives of DL, many architectures reach out in popularity. Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) are two of them, which have the biggest impact within the field of health informatics. The details configurations of both CNNs and RNNs are explained next.

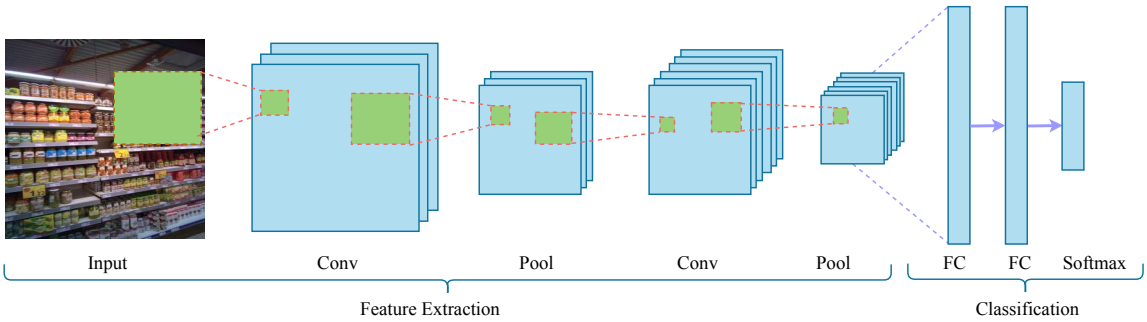


Figure 2.1: The basic architecture of a CNNs.

2.2 Convolutional Neural Network

This network formation was first introduced by Fukushima in 1988 (Fukushima, 1988). It was not broadly used due to lack of computation hardware for training the network. LeCun et al. implemented a gradient-based learning algorithm to CNNs and achieved prominent results for solving the problem of handwritten digit recognition in 1990 (LeCun et al., 1998). Afterwards, researchers improved CNNs models and reported state-of-the-art results in various classification tasks. Currently, CNNs has studied as the common extensively used method of DL; particularly in vision-based applications and has presented state-of-the-art outcomes in DL related tasks. However, CNNs controls both, useful feature extraction and clear discrimination capability. Hence, it is widely applied at feature extraction or generation and model selection steps in a DL system. A conventional CNNs architecture commonly includes intermittent layers of convolution and pooling accompanied by one or more FC layers at the end. A common system architecture of a CNNs system is shown in Figure 2.1. It mainly consists of two major parts including feature extractors and a classifier. In the feature extraction layers, every layer of the network gets the output from its immediate earlier layer as its input and transfers its output as the input to the next layer. This layer is made of several convolutions and pooling layers. Every node of the convolution layer obtains the features from the input images by convolution processes on the input nodes and the pooling layer helps to reduce the dimension of the generated features propagates to heights level. The output feature map of the final layer of the CNNs is used as

the input to FC layer which is called classification layer. In the classification layer, the final output of feature extraction layers is chosen as inputs corresponding to the dimension of the weight matrix of the final neural network. In some instances, the FC layer is replaced with the layer of global average pooling. Apart from separate learning stages, several supervisory units such as batch normalization and dropout are also included to optimize CNNs performance (Bouvrie, 2006). The combination of CNNs layers and units performs a leading role in creating novel architectures for obtaining enhanced performance. This section briefly explains the role of these layers and units in CNNs architecture.

2.2.1 Convolutional Layer

The Convolutional layer is designed by a collection of convolutional kernels (every neuron performs as a kernel). These kernels are correlated with a small region of the image recognized as a receptive field. It acts by splitting the image into little blocks (known as a receptive field) and convolving them with a particular set of weights (multiplying elements of the filter (weights) with equivalent receptive field elements (Bouvrie, 2006). Convolution procedure is described in the equation,

$$C_l^k = (I_{x,y} * K_l^k), \tag{2.1}$$

where $I_{x,y}$ defined the input image and x, y indicates the spatial locality where K_l^k presented l^{th} convolutional kernel of the k_{th} layer. Splitting image into little

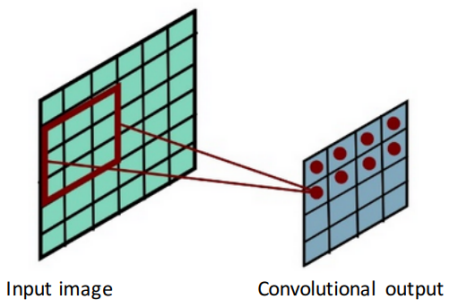


Figure 2.2: The fundamental operation of a Convolution layer, filters work on each part of the image, therefore, they are seeking for the equivalent feature everywhere in the image.

blocks assists in extracting locally correlated pixel values. This locally aggregated information is also recognized as feature patterns. By sliding the convolutional kernel on the image obtained the various set of features with the same set of weights. Convolution procedure may additionally be classified into many kinds based on the type and size of filters, the type of padding, and the orientation of convolution (LeCun et al., 2015). An example of the convolution operation is shown in Figure 2.2.

2.2.2 Pooling Layer

The output of the convolution operation is obtained features from the input image. The pooling layers analyze these features collected by the convolutional layer and create a compressed version of the features enclosed in them. Pooling layer also called downsampling layer which is aggregates similar information in the neighbourhood of the receptive field and outputs the highest response within this local region (Lee et al., 2016):

$$P_l = f_p(C^l_{x,y}). \tag{2.2}$$

The pooling or downsampling operation shows in Equation 2.2, where $C^l_{x,y}$ and P_l defines l^{th} output and input feature map, f_p represents the type of pooling operation. The application of the pooling method helps to obtain a combination of features, which are invariant to translational changes and little distortions (Scherer et al., 2010; Marc'Aurelio Ranzato et al., 2007). Moreover, pooling can also help in improving

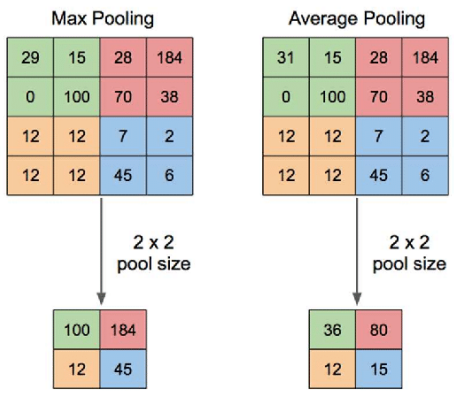


Figure 2.3: Examples of max and average pooling operations.

the generalization by decreasing overfitting. Including this, decrease in the size of feature maps controls the complexity of the network. There are several kinds of pooling operations such as max, mean, average, and L2 pooling are commonly utilized for extracting translational invariant features (Wang et al., 2012; Boureau et al., 2010). Examples of max and average pooling operation is shown in Figure 2.3.

2.2.3 Fully Connected Layer

Fully connected (FC) layers also known as classification layer is commonly used at the end of the network for the classification task. The FC layers calculate the score of every class from the extracted features from a convolutional layer in the previous steps. The final layer feature maps are represented as vectors with scalar values which are transferred to the FC layers. More generally, it receives input from the earlier layer and analyses output of all earlier layers globally (Lin et al., 2013). It creates a non-linear combination of selected features that are applied for the classification of data. The fully connected layers are used as a soft-max classification layer. There are no specified rules on the number of layers which are included in the network model. Nevertheless, in most scenarios, two to four layers have been introduced in different architectures including LeNet (LeCun et al., 1998), AlexNet (Krizhevsky et al., 2012), and VGG Net (Simonyan and Zisserman, 2014). In terms of computation, the FC layers are very costly. For this reason, different methods have been introduced during the last few years, including the average pooling layer and global average pooling layer which help to decrease the number of parameters in the network significantly.

2.3 Advanced Training Methodologies

There are several advanced methods to use for perfectly train a DL model. The procedures including data pre-processing, a suitable initialization technique, batch normalization, advanced activation functions, alternative pooling methods, network regularization techniques, and better optimization method for training. The following sections are reviewed on different advanced training methods.

2.3.1 Data Pre-processing

Currently, different procedures have been used before feeding the data to the network. The procedure is called "data pre-procession" or "data augmentation" to prepare a dataset for training the DL models. There are several techniques are available for data pre-processing, including; random cropping, flipping data concerning the horizon or vertical axis, sample re-scaling, shearing, refection, mean subtraction, color jittering, principal component analysis (PCA) whitening, and so on (Mikołajczyk and Grochowski, 2018).

2.3.2 Network Initialization

The initialization of DL models has a significant impact on overall network performance. Earlier, the networks were commonly initialized with random weights. Training large DNNs models with huge dimensionality data has become a difficult task because the weights of the network should not be symmetrical due to the back-propagation process. Hence, effective initialization methods are necessary for training this type of DNNs. Since the last few years, several efficient network initialization techniques have been proposed. LeCun (LeCun et al., 2012) introduced simple and efficient techniques. In their approaches, the weights are scaled via the inverse of the square root of the number of input neurons of the layer, which can be defined by $1/\sqrt{N_l}$, where N_l is the number of input neurons of l^{th} layer. Another well-known initialization approach is "Xavier" based on the symmetric activation function related to the hypothesis of linearity (Glorot and Bengio, 2010). Recently, a popular initialization technique has proposed by Kiming He in 2015 (He et al., 2015a). In this method, the weights are initialized keeping in mind the size of the earlier layer which supports in achieving a global minimum of the cost function faster and more effectively. The weights distribution of l^{th} layer will be a normal distribution with mean zero and variance $\frac{2}{n_l}$ which can be represented as follows:

$$W_l \sim \mathcal{N}(0, \frac{2}{n_l}). \tag{2.3}$$

Dmytro M. et al. presented a Layer-sequential unit-invariance(LSUV) (Mishkin and Matas, 2015) initialization technique that is a data-driven and achieved excellent recognition performance on many benchmark datasets including ImageNet.

2.3.3 Batch Normalization

Batch normalization is applied to solve the problems associated with internal covariance shift inside the feature map. The internal covariance shift is a variation in the distribution of hidden units' values that makes slower the convergence by forcing learning rate to a small value, which demands precise initialization of parameters. Batch normalization for transformed feature map A_l^k is shown in Equation (2.4):

$$B_l^k = \frac{A_l^k}{\sigma^2 + \sum_i A_i^k}, \tag{2.4}$$

where B_l^k defines normalized feature map, A_l^k is the input feature map and σ describes variation in feature map. Batch normalization unifies the distribution of feature map values by normalizes the activations at each batch and maintains zero mean and unit variance (Ioffe and Szegedy, 2015). It allows us to apply enough higher learning rates and be less careful about initialization during the training of a very deep network. Moreover, it improves the gradient flow smoothly and plays as a regulating agent, which enhances the generalization of the network without relying on dropout.

2.3.4 Activation Function

The activation function helps to introduce non-linearity in the modelling abilities of the network and serves as a decision function that boosts in learning a complex pattern. Choice of a suitable activation function can accelerate the learning procedure. Activation function for convolved feature map is represented in Equation (2.5):

$$A_l^k = f_A(C_l^k), \tag{2.5}$$

where C_l^k is the output of convolution operation that is assigned to activation function $f_A(.)$ which combines non-linearity and produces a transformed output A_l^k for k_{th} layer. Thus, several activation functions such as Sigmoid, TanH, Rectified Linear Units (ReLU), and variations of ReLU such as Leaky ReLU, Exponential Linear Unit (ELU), and Parametric Rectified Linear Unit (PReLU) are described in literature (Nair and Hinton, 2010; Xu et al., 2015a; Gu et al., 2018; Nwankpa et al., 2018) are applied to instill nonlinear combination of features. Figure 2.4 shows examples of the activation functions. ReLU and its variations are preferred over other activation functions as it tackled with the vanishing gradient problem (Nair and Hinton, 2010).

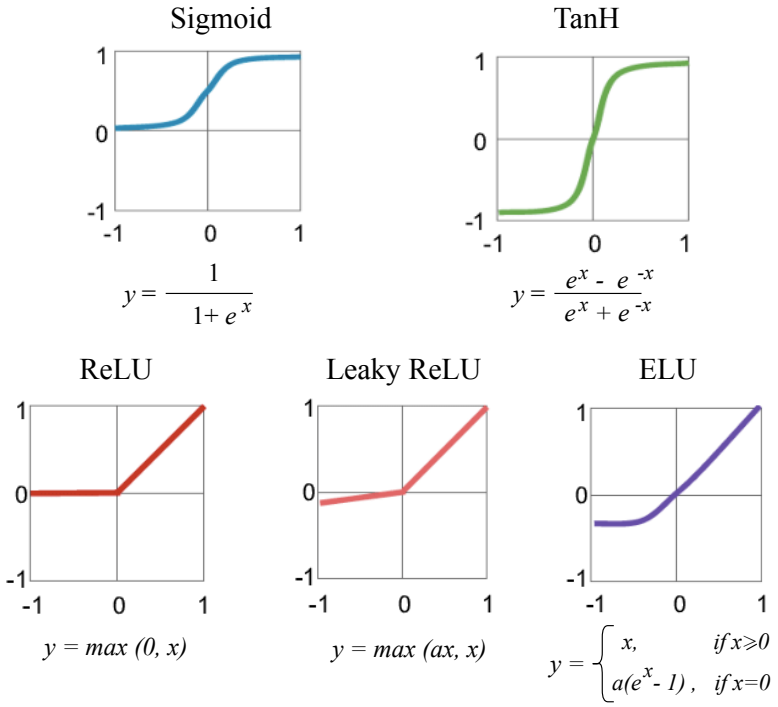


Figure 2.4: Examples of activation functions.

2.3.5 Dropout

Dropout proposes regularization within the network, which finally enhances generalization by randomly skipping a few units or connections with a particular probability. In NNs, many connections that learn a non-linear association are seldom co-adapted, which produces overfitting. This random dropping of some connections

or units provides many thinned network structures out of which one typical network is chosen with small weights. This picked architecture is then regarded as an approximation of all of the proposed networks (Srivastava et al., 2014). Figure 2.5 shows the concept of the Dropout. Dropout randomly drops neurons from a network while training. Empirically, this method often gives powerful regularization for network training.

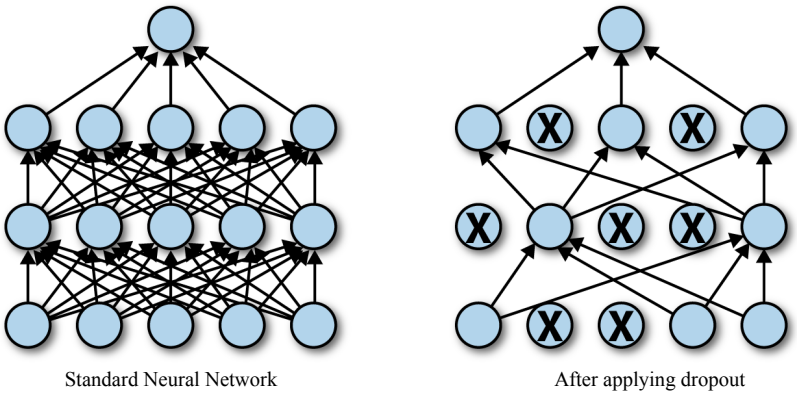


Figure 2.5: Graphical illustration of the Dropout concept.

2.3.6 Special Pooling Layer

The common pooling layers are average, max and mean pooling. There is also a special type of pooling operations introduced including pyramid pooling (He et al., 2015b), multi-scale pyramid pooling (Yoo et al., 2015). In (Graham, 2014) presented a new architecture with Fractional max pooling, which gives state-of-the-art classification accuracy on the CIFAR-100 and CIFAR-10 dataset. Different types of pooling approaches are reviewed in the literature (Lee et al., 2016) including mixed, gated, and tree as a generalization of pooling functions.

2.3.7 Optimization Techniques

Several optimization approaches are available for training DL models including SGD, Adagrad, AdaDelta, RMSprop, and Adam (Ruder, 2016). The activation functions have been improved by adding variable momentum with SGD which significantly increased network performance during training and testing. The main contribution

of Adagrad was to calculate the adaptive learning rate during training. For this process, the summation of the gradient magnitude is estimated to calculate the adaptive learning rate. The summation of gradient magnitude become large with the big number of training epochs. As a result, the learning rate decreases entirely that induces the gradient to approach zero promptly. The major disadvantage of this method is that it creates difficulties during training. Following, RMSprop was introduced regarding only the gradient magnitude of the immediately earlier iteration, which prevents the problems with Adagrad and gives more reliable performance in some situations. The Adam optimization method is introduced based on the momentum and the gradient magnitude for estimating adaptive learning rate as like RMSprop. Adam has increased overall accuracy and supports for efficient training with the better convergence of deep learning methods (Le et al., 2011). Recently, the improved version of Adam optimization has been introduced and gives better performance with fast and accurate convergence (Koushik and Hayashi, 2016).

2.4 Advanced CNNs Architectures

Several popular state-of-the-art CNNs architectures are discussed in this section. Generally, most of the CNNs architectures consist of stacks of several convolutional and pooling layers followed by a FC layer and softmax layer at last. For examples, LeNet (LeCun et al., 1998), AlexNet (Krizhevsky et al., 2012)and VGG Net (Simonyan and Zisserman, 2014) are composed by the above concept of CNNs architecture. Many advanced efficient architectures have been introduced such as GoogLeNet with Inception module (Szegedy et al., 2016, 2017), Residual Networks (He et al., 2016). The details about some common networks that we used for this research are present in the next sections.

2.4.1 VGGNet

The runner-up at the ILSVRC 2014 contest is entitled VGGNet by the community and was developed by the Visual Geometry Group (VGG), from University of

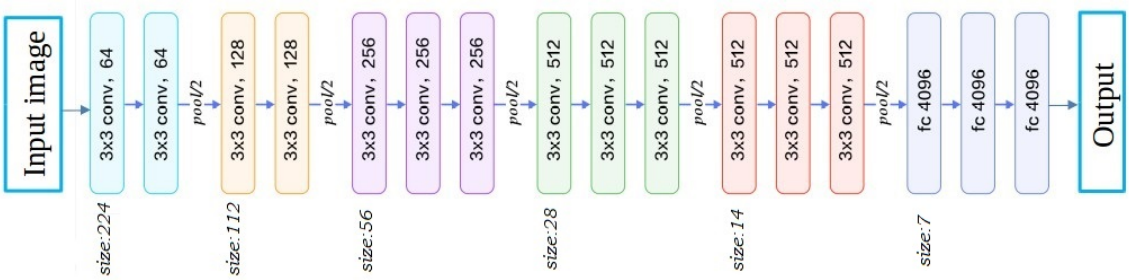


Figure 2.6: Graphical representation of VGGNet.

Oxford (Simonyan and Zisserman, 2014). VGGNet composed of 16 convolutional layers and is attractive of its uniform architecture. The key contribution of this work is that it proves that the depth of a network is a significant part to achieve better classification accuracy in CNNs. The VGGNet architecture used a sequence of two convolutional layers with ReLU activation function following by a single max-pooling layer. Afterwards, several fully connected layers also using a ReLU activation function. Finally, a Softmax layer used for the computes the final classification score. Three different versions of VGGNet were proposed, VGG-11, VGG-16 and VGG-19 with 11,16, and 19 layers respectively. The weight configuration of the VGGNet is publicly available and has been utilized in many different applications and challenges as a baseline feature extractor. However, VGGNet contained 138 million parameters, and VGG-19 had 16 convolution layers, the most computational expensive model. Figure 2.6 shows the architecture of VGGNet.

2.4.2 GoogleNet

The winner of ILSVRC 2014 competition, GoogleNet, proposed by Christian Szegedy from Google (Szegedy et al., 2016). The main contribution of GoogleNet, the initial version known as "Inception-V1"‘, was reducing computation complexity compared to the traditional CNNs. The architecture was introduced "Inception Module", whereby it includes multi-scale convolutional transformations applying split, transform and merge idea for feature extraction. The initial concept of the Inception layer can be seen in Figure 2.7. This block encapsulates filters of different sizes (1×1, 1×1, and 1×1) to capture spatial information in aggregation with channel information

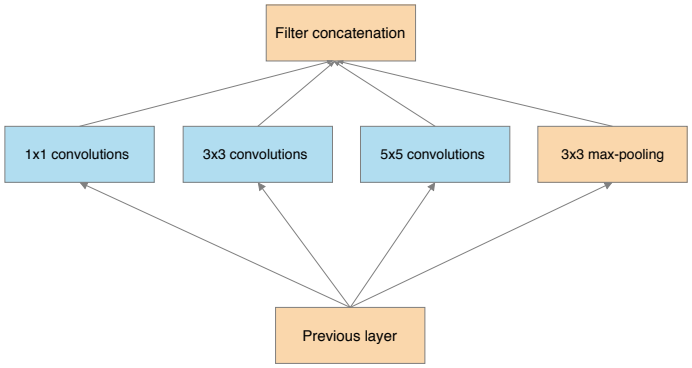


Figure 2.7: Naive version of Inception Module.

at different spatial resolutions. GoogleNet, "Inception-V1", achived state-of-the-art classification accuracy using a sequence of Inception module shown in Figure 2.8.

The variation between the initial or naive inception module and final inception module was the addition of 1×1 convolution kernels. These 1×1 convolution kernels are supported for dimensionality reduction before computationally expensive layers. GoogNet introduced very deep network, "Inception-V2", with 22 layers in total, which is very deep and compare to other networks before it. GoogleNet had also very fewer (7M) network parameters which was much lower than its predecessor AlexNet(60M) and VGGNet(138M). GoogleNet also presented the idea of auxiliary learners to speed up the convergence rate. Inception-V3 or V4 and Inception-ResNet-V2, which are an upgraded version of Inception-V1 or V2. The concept of Inception-V3 was to decrease the computational cost of deep networks

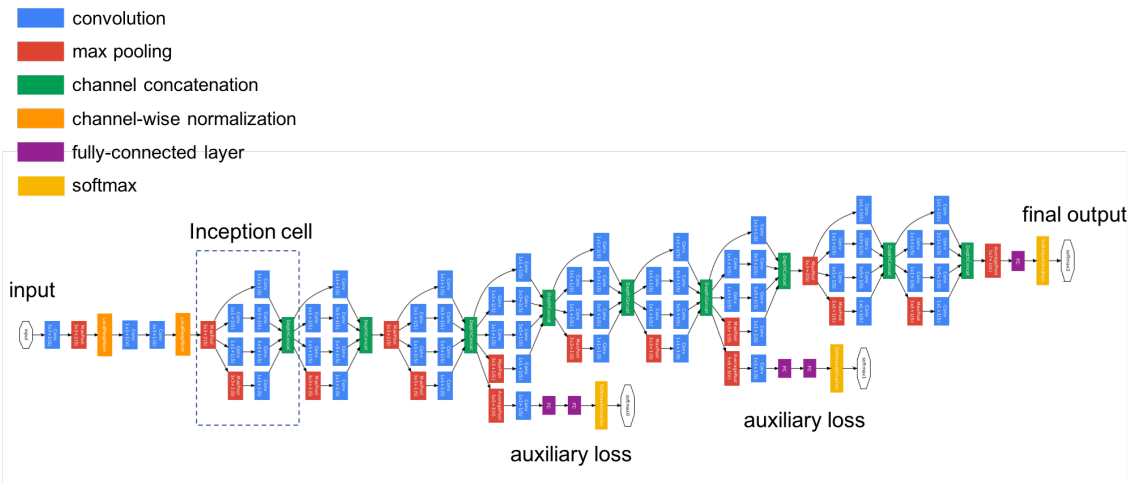


Figure 2.8: Graphical representation of GoogleNet (Inception-V1).

without changing the generalization. Therefore, Szegedy et al. (Szegedy et al., 2016) replaced large size filters (5×5 and 7×7) with small and asymmetric filters (1×7 and 1×5) and used 1×1 convolution as a bottleneck before the large size filters. This performs the conventional convolutional operation similar to cross-channel correlation. Inception-ResNet-V2 joined the power of residual learning and inception module (Szegedy et al., 2017). The Inception-V4 with residual connections (Inception-ResNet-V2) represents that training with residual connections accelerates the training of Inception networks significantly.

2.4.3 ResNet

Residual Network was the winner of ILSVRC 2015 competition, named ResNet (He et al., 2016), proposed by Kaiming He et al introduced a novel architecture with "skip connections" and features large batch normalization. For the first time, Residual Network designed with ultra-deep networks that did not suffer from the vanishing gradient problem that previous models had. Such skip connections are also known as gated units or gated recurrent units and have a powerful relationship to recent successful components utilized in RNNs. It achieves a top-5 error rate of 3.57% which overcomes human-level performance on ImageNet dataset. Figure 2.9 shows the diagram of residual unit. ResNet is developed with multiple numbers of layers: 34, 50,101, and 152. The common ResNet-50 model included 49 convolution layers and 1 fully connected layer at the end of the network. The number of total weights

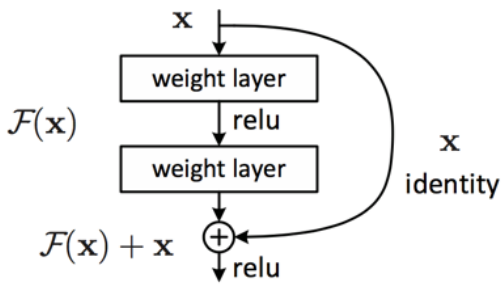


Figure 2.9: The diagram of Residual unit.

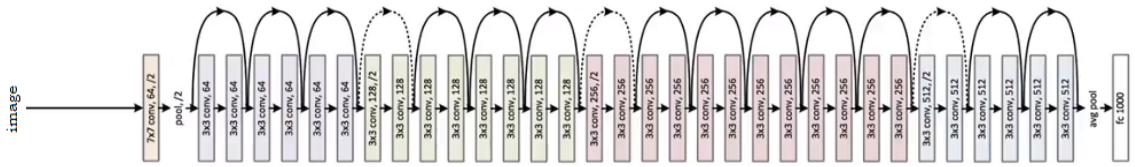


Figure 2.10: The architecture of ResNet.

for the entire network is 25.5M. Figure 2.10 illustrates a very deep ResNet with sequences of Residual unit.

2.5 Recurrent Neural Networks (RNNs)

Traditionally, RNNs were developed for the analysis of discrete sequences. On the other hand, they take as their input not only the current input example they see, but also what they have perceived previously in time. In a classification framework, the model learns a distribution over classes $P(y \mid x_1, x_2, ..., x_T; \Theta)$ given a sequence $x_1, x_2, ..., x_T$, instead of a single input vector x. The traditional RNNs keeps a latent or hidden state h at time t that is the output of a non-linear mapping from its input x_t and the previous state h_{t1}:

$$ht = \sigma(W_{xt} + R_{ht1} + b), \tag{2.6}$$

where the weight matrices W and R are shared additionally. For the classification, one or more fully connected layers are commonly combined followed by a softmax layer to map the sequence to a posterior over the classes:

$$P(y \mid x_1, x_2, ..., x_T; \Theta) = softmax(h_T; W_{out}, b_{out}) \tag{2.7}$$

Figure 2.11 shows different types of RNNs. Since the gradient requires to be backpropagated from the output through time, RNNs are inherently deep (in time) and consequently suffer from the same problems with training as regular deep neural networks (Bengio et al., 1994). To this end, different specialized memory units have been introduced, the earliest and most famous one is the Long Short-Term Memory

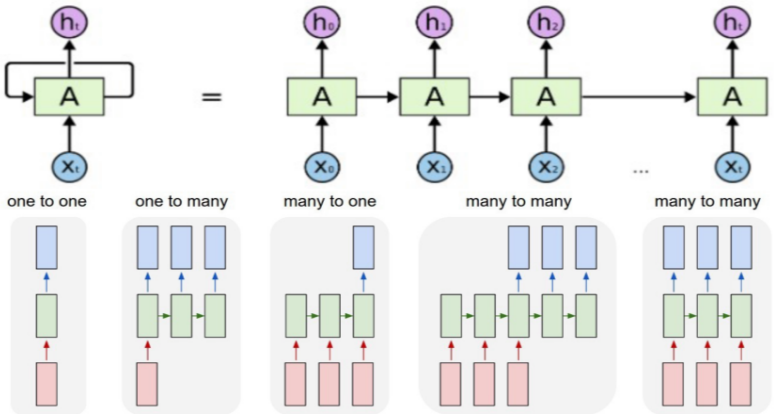


Figure 2.11: Examples of RNNs models.

(LSTM) cell (Hochreiter and Schmidhuber, 1997). A recent simplification of the LSTM, the Gated Recurrent Unit (GRU) is also widely used. Although initially RNNs are proposed for one-dimensional input but currently its increasingly applied to images also. For examples, "pixelRNNs" (Oord et al., 2016) are used as autoregressive models, generative models that can eventually produce new images similar to samples in the training set.

2.6 Generative Adversarial Networks (GANs)

In 2014, Ian Goodfellow introduced a revolutionary idea (Goodfellow et al., 2014), make two neural networks compete (or collaborate, it is a matter of perspective) with each other. One neural network tries to generate realistic data, named "generator" (remarks, GANs can be used to model any data distribution, but are largely used for images nowadays), and the other network tries to discriminate, named "discriminator", between real data and data generated by the generator network. The generator network uses the discriminator as a loss function and updates its parameters to generate data that starts to look more realistic. Figure 2.12 shows the basic concept of a GANs model. In the past few years, GANs have been widely studied. Arguably the revolutionary methods are in the area of computer vision such as apparent image generation, image to image translation, facial attribute manipulation and similar domains. Nevertheless, the significant success

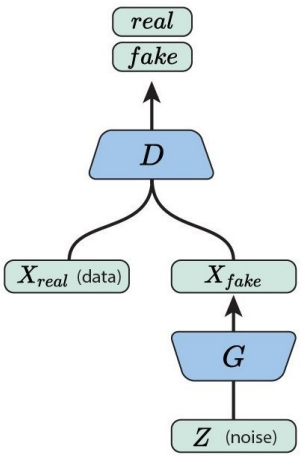


Figure 2.12: The basic architecture of a GANs model.

obtained in the computer vision field by implementing GANs to real-world problems. Several GANs are introduced, such as Deep convolutional generative adversarial networks (DCGANs) (Radford et al., 2015), Conditional GANs (cGANs) (Isola et al., 2017), stacked Generative Adversarial Networks (StackGANs) (Zhang et al., 2017), InfoGANs (Chen et al., 2016b), Wasserstein GANs (WGANs) (Arjovsky et al., 2017) and so on. DCGANs is an improvement of GANs. It is more stable and generates higher quality images. In DCGANs, batch normalization is used in both networks (the generator and the discriminator network). DCGANs can be applied for style transfer. For example, the model can use a dataset of handbags to generate shoes in the same style as the handbags. In conditional GANs (cGANs) used the same DCGANs and imposed a condition on both generator and discriminator network inputs. The condition should be in the form of a one-hot vector of the input. This is correlated with the image to Generator or Discriminator as real or fake. StackGANs proposed to solve the problem of synthesizing high-quality images from text descriptions. StackGANs generated 256×256 photo-realistic images conditioned on text descriptions. The model decomposes the difficult problem into more controllable sub-problems through a sketch-refinement method. InfoGANs is an information-theoretic continuation to the GANs that can learn disentangled representations in an unsupervised method. It is inspired by the desire to disentangle and control the features in generated images. It includes the addition of control

variables to generate an auxiliary model that predicts the control variables, trained via mutual information loss function. InfoGANs are used when the dataset is very complex, not well labelled. WGANs modify the loss function to combine a Wasserstein distance. WGANs add some tricks to allow discriminator network to approximate Wasserstein distance between real and model distributions. Wasserstein distance approximately indicates "how much work is required to be done for one distribution to be adapted to match another" and is remarkable in a way that it is defined even for non-overlapping distributions.

Chapter 3

FoodPlaces: Learning Deep Features for Food Related Scene Understanding

3.1 Introduction

Obesity, diabetes and heart diseases are counted among the problems caused by the irregular diet and bad nutrition intakes throughout the world. It is critical to detect bad nutrition intakes based on the food-related environment to point out unhealthy eating habits. Therefore, an automated food-related environment recognition system is needed, which can identify the food intakes in public environments such as, bars, restaurants and cafeterias. To date, numerous methods have been proposed in the area of food item detection and recognition. A food recognition system based on the hand-crafted features using the traditional computer vision algorithms is proposed in (Yang et al., 2010; Matsuda and Yanai, 2012). These algorithms are based on comparative spatial interactions of local descriptors, the fusion of features after extraction, graph-based ranking algorithm and co-occurrence matrices among the food items. They have large scale, low adjustment rate to big scale, which leads to high computational cost. Currently, several machine learning approaches are used for precise food items recognition. A food classification method using mine discriminative elements with random forest (RF) was proposed by Bossard et al. (Bossard et al., 2014), which was tested on the Food 101 test database. In

this work, RF is used to cluster graph-based super-pixels to train the model. In this work, the authors have also exploited some other classification methods such as, improved Fisher vectors (IFV) (Sánchez et al., 2013) and randomized clustering forests (RCF) (Moosmann et al., 2008).

Recently, a modern deep learning-based convolutional neural networks (CNNs) approach is used by Mayers et al. in the application of food recognition (Meyers et al., 2015). In this work, GoogleNet Inception-V1 deep learning model was used on five different datasets. Later on, Liu et al. used the modified Inception model on UEC-256 and Food-101 food datasets in the food classification problem and generated acceptable results (Liu et al., 2016). Moreover, deep learning models are also used in different mobile based applications such as, "Snap-n-Eat" (Zhang et al., 2015), "Lose it" (Dohan and Tan, 2011) and "Platemate" (Noronha et al., 2011) to detect and recognize different types of food items. However, no work has been done in the area of scene recognition related to the food environment. A novel scene classification method using CNNs deep features was proposed by Bolei Zhou et al. (Zhou et al., 2014) to compare the density and diversity of images using the "Places" dataset. However, this method is proposed to classify the scenes based on different places such as an airfield, art studio, bathroom, classroom, etc. In this work, we propose a deep learning-based food-related scene recognition system to classify different types of food places by using state-of-the-art CNNs models. We have used three different models: VGG-16 (Simonyan and Zisserman, 2014), ResNet-50 (He et al., 2016) and Inception-V3 (Szegedy et al., 2016). The classification results show that the Inception-V3 model yields the highest accuracy among the used state-of-the-art models.

3.2 Proposed Approach

3.2.1 Convolutional Neural Networks

The early fruitful application of Convolutional Networks was developed by Yann LeCun (LeCun et al., 1998), called LeNet that was applied to read digits and recognize hand-written numbers on checks by several banks and zip codes. For image classification and recognition CNN's become more powerful after introducing AlexNet by Krizhevsky et al. (Krizhevsky et al., 2012), which won the first prize in the ImageNet Large Scale Visual Recognition Competition (ILSVRC) 2012 competition. Consequently, in the history of CNNs evolution there are different breakthroughs: VGGNet (Simonyan and Zisserman, 2014) by Simonyan et al., GoogleNet (Szegedy et al., 2016) by Szegedy et al and ResNet (He et al., 2016) by He et al. VGGNet architecture is very popular for its simplicity and depth. It yields the best error rate of 7.3% using ImageNet. It has 19 convolutional layers that exactly used 3x3 filters with stride 1 and pad 1, and the activation function is 'ReLU', as well as 2x2 max-pooling layers with stride 2. VGGNet reinforced the perception that CNNs have to have a deep network of layers to work. He et al. introduced ResNet in (He et al., 2016)] and it is a version of 'tropical architecture' that depends on micro-architecture modules which are also named 'network-in-network architectures'. The term micro-architecture mentioned to the group of 'building blocks' was used to develop the network. The ResNet architecture illustrated that very deep networks can achieve satisfactory results being trained by standard stochastic gradient descent (SGD) through the residual modules. However, ResNet is much deeper than VGG-16 but the model size is significantly smaller for the usage of global average pooling rather than fully-connected layers. ResNet-50 attained 24.7% top-1 and 7.8% top-5 error for 1-crop validation error of ImageNet.

In the year of 2014 GoogLeNet won the ILSVRC-14 competition. The proposed GoogLeNet improved the top-5 error of ImageNet from 16.4% to 6.7% which achieved by AlexNet. Here GoogleNet was introduced 'Inception'- a deep convolutional neural network architecture. This model shows better performance of fewer parameters

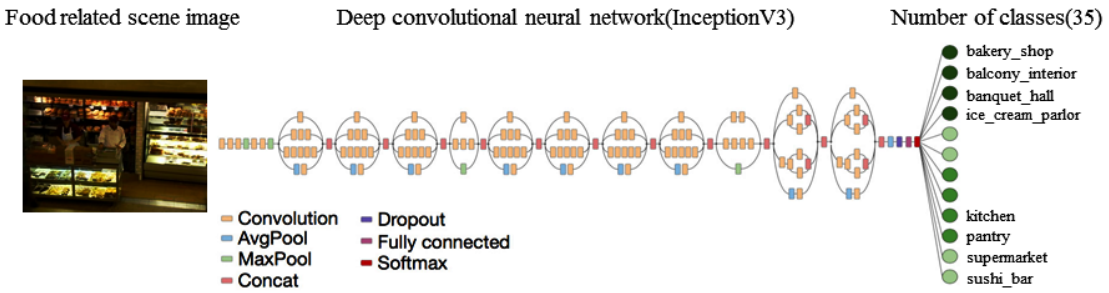


Figure 3.1: Inception-V3 model architecture for the food-related scene classification.

(4 million) as compared to AlexNet with 60 millions (Szegedy et al., 2016). It has 22 layers deep network and to ignore the irrelevant parameters the inception network used an average pooling instead of fully connected (FC) layers at the top of the ConvNet. It also introduced extra losses fixed to the classification error of the intermediate layers. Subsequently, there are various additional progressive versions of the Inception models. In the Inception-V2 (Szegedy et al., 2016) introduced a "Batch Normalization" calling the CNNs as BN-Inception. Afterwards, in the third emphasis, the model architecture performance was increased by adding factorization concept that was referred to as Inception-V3. This network has much more advantages compared to other previous network models. It achieves 21.2% top-1 and 5.6% top-5 error for single frame evaluation. Finally, it proved very good classification performance at modest computational cost, specifically, the addition of residual connections contributed to advance the training significantly. Figure 3.1 shows the Inception-V3 model architecture for our 35 food-related scene classification task. We modified it by removing the intermediate auxiliary logits output to adopted with our problem.

3.2.2 Fine-tuning of CNNs for Food Related Environment Classification

In this work, we describe the fine-tunning process of several CNNs state-of-the-art models to adapt them specifically to the food-related scene classification using the new dataset described in Section 3.3.1. Moreover, we discuss how we extract the deep

Table 3.1: Description of the proposed "FoodPlaces" dataset.

Dataset		Original no. of classes	Original no. of images	No. of Food related scene classes	No. of Food related images
FoodPlaces	Places365	365	1,946,750	35	176,517
	ImageNet	1000	14,197,122	26	30,600
	SUN397	397	108,754	30	11,372

learning features for upgrading their perspective fitness. We applied the fine-tuning method to our proposed network structure which is using the pre-trained model as a checkpoint and progress to training the neural network.

ImageNet is a rich dataset that contains a huge number of images mainly used for object classification. Its pre-trained model is also well-trained; therefore, it can be used to classify different types of images in different applications. In this work, we have used three different pre-trained models from ImageNet: VGG-16, ResNet-50 and Inception-V3. For these models, we loaded a checkpoint that stored all the tensors after approximately two weeks of fine-tuning them on our FoodPlaces dataset. The last layer of the network classifies the classes of the food-related scene images.

3.3 Experimental Evaluation

3.3.1 The FoodPlaces Database

In the deep learning method, it is necessary to have a big image dataset for learning deep features. We propose a new dataset by combining three public datasets "Places365" (Zhou et al., 2016), "ImageNet" (Russakovsky et al., 2015b) and "SUN397" (Xiao et al., 2010) dataset. We named it "FoodPlaces". A detailed description of the proposed dataset is shown in Table 3.1.

In this dataset, we selected 35 common food-related scenes from the above mentioned public datasets. The collected images from those datasets have large scene taxonomy, which contains rich classes to cover the various visual surroundings of our daily life experience. Figure 3.2 shows the food-related scene images from the proposed database.

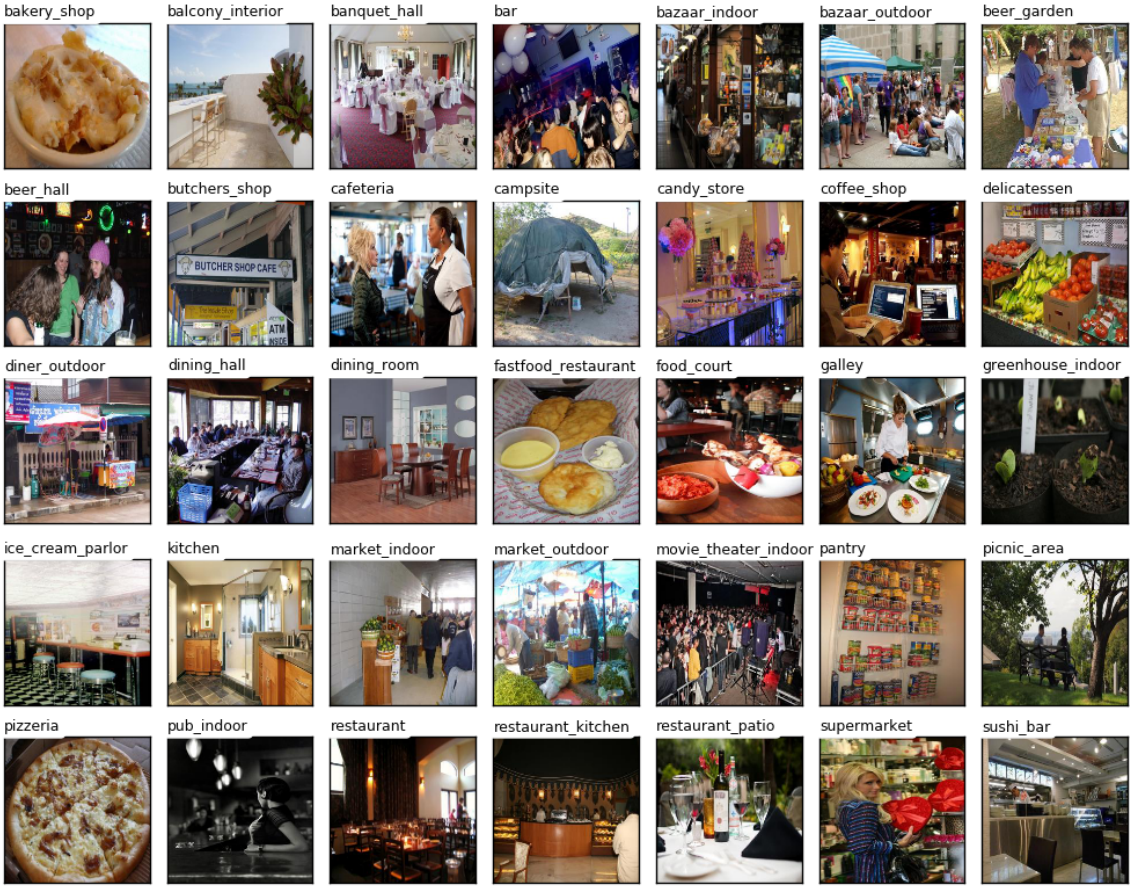


Figure 3.2: Examples of images from the food-related scene classes.

Table 3.2: Selected classes from public datasets

Dataset	Classes
Places365	bakery_shop, balcony_interior, banquet_hall, bar, bazaar_indoor, bazaar_outdoor, beer_garden, beer_hall, butchers_shop, cafeteria, campsite, candy_store, coffee_shop, delicatessen, diner_outdoor, dining_hall, dining_room, fastfood_restaurant, food_court, galley, greenhouse_indoor, ice_cream_parlor, kitchen, market_indoor, market_outdoor, movie_theater_indoor, pantry, picnic_area, pizzeria, pub_indoor, restaurant, restaurant_kitchen, restaurant_patio, supermarket, sushi_bar.
ImageNet	bakery, banquet_hall, bar, bazaar, beer_garden, beer_hall, butchers_shop, cafeteria, campsite, candy_store, coffee_shop, delicatessen, diner, dining_hall, dining_room, fastfood, food_court, greenhouse, kitchen, market, pantry, picnic_area, pizzeria, restaurant, supermarket, sushi_bar.
SUN397	bakery_shop, balcony_interior, banquet_hall, bar, bazaar_indoor, bazaar_outdoor, butchers_shop, cafeteria, campsite, candy_store, coffee_shop, delicatessen, diner_outdoor, dining_room, fastfood_restaurant, food_court, galley, greenhouse_indoor, ice_cream_parlor, kitchen, market_indoor, market_outdoor, movie_theater_indoor, pantry, pub_indoor, restaurant, restaurant_kitchen, restaurant_patio, supermarket, sushi_bar.

Table 3.2 shows the name of the selected classes from the above-mentioned dataset. The Places365 dataset is used for the scene classification. It has 365

scene classes. It contains images of different categories concerning a scene or place name as opposed to an object label. Among these classes, we found only 35 classes are the food-related scene. We selected the classes based on the maximum possibilities of food appearance or peoples eating places. The quality of images of Places365 are high-resolution and have been resized to 256×256 despite their original aspect ratios and the details in (Zhou et al., 2016). On the other hand, ImageNet dataset is mainly used for object classification and recognition. The number of classes is in ImageNet is 1000 and the size of the images are different from each other (Russakovsky et al., 2015b). However, it has some classes which can be used for the scene classification. We selected 26 of them for our dataset, which also similar class name with the Places365. Another, the SUN397 dataset was created by a quasi-exhaustive list of scene classes with various functionalities, specifically classes with particular identities in discourse. It has also 397 scene classes and the size of images are different from each other also. The details are in (Xiao et al., 2010). We selected 30 classes from it and those are also similar to the Places365. The maximum number of food-related scene classes are in Places365, so we chose those name as a baseline for our FoodPlaces dataset. In ImageNet, the class name bakery, bazaar, dinar, fast food, greenhouse and market are similar to the class name bakery_shop, bazaar_indoor, bazaar_outdoor, diner_outdoor, fastfood_restaurant, greenhouse_indoor, market_indoor and market_outdoor in Places365 and SUN397. We copied manually images related to FoodPlaces class name from the similar classes of ImageNet. Also, seven classes named; balcony_interior, galley, ice_cream_parlor, movie_theater_indoor, pub_indoor, restaurant_kitchen and restaurant_patio do not exist in ImageNet and five classes; beer_garden, beer_hall, dining_hall, picnic_area and pizzeria do not exist in SUN397. In total, FoodPlaces contains more than two hundred thousand images comprising of 35 food-related scenes and each class contains a minimum of 5000 images for training and test.

3.3.2 Methodology

In the initial stage of the experimental setup, we divided our dataset for the training and test phase. The dataset was split into 80% for training and the rest of the images for testing. Here we utilized NVIDIA GTX 1070 with 1920 CUDA cores and 8GB memory size which facilitated to run a complex network architecture. The scheme for deep learning is the latest version of Keras 2.0.3 (Chollet et al., 2015) with Tensorflow backend. In the ImageNet pre-trained model classifier, the number of classes is 1000, but we have only 35 food-related scene classes. So we have to tune the dimension of the last fully-connected (FC) layer of the networks with the number of 35 classes. We have tested the default model on FoodPlaces and also retrained the final layer to get a model depending on the pre-trained model. We found that the default and the retrained last layer of CNNs models cannot produce performance higher than 40% and 60% respectively. To overcome this problem, we retrained the full models until the softmax layers of our pre-trained models and got much higher accuracy. It is necessary to fit and fine-tune the optimization parameters, for example, weight decay, which avoids the overfitting problem and helps to provide the balance in between variance and bias. In the training process, we used momentum (Sutskever et al., 2013) with a decay of 0.9, while our best models were achieved using Adadelta (Zeiler, 2012) with rho=0.95, epsilon=1e-06 and decay of 0.9. We used a learning rate of 0.001, that decayed every ten epochs using an exponential rate of 0.001 as well as batch size 64 is also given to the network. Model evaluations are performed using a running average of the parameters computed over time.

3.3.3 Experimental Results

The selection of neural network models decides the final prediction result after the confluence of retraining the network. In this section, we show the experimental results obtained on the "FoodPlaces" dataset to classify the food-related environment. To retrain all the models, a random number of images are selected from our database.

Figure 3.3, 3.4 and 3.5 compares three of the best state-of-the-art CNNs models:

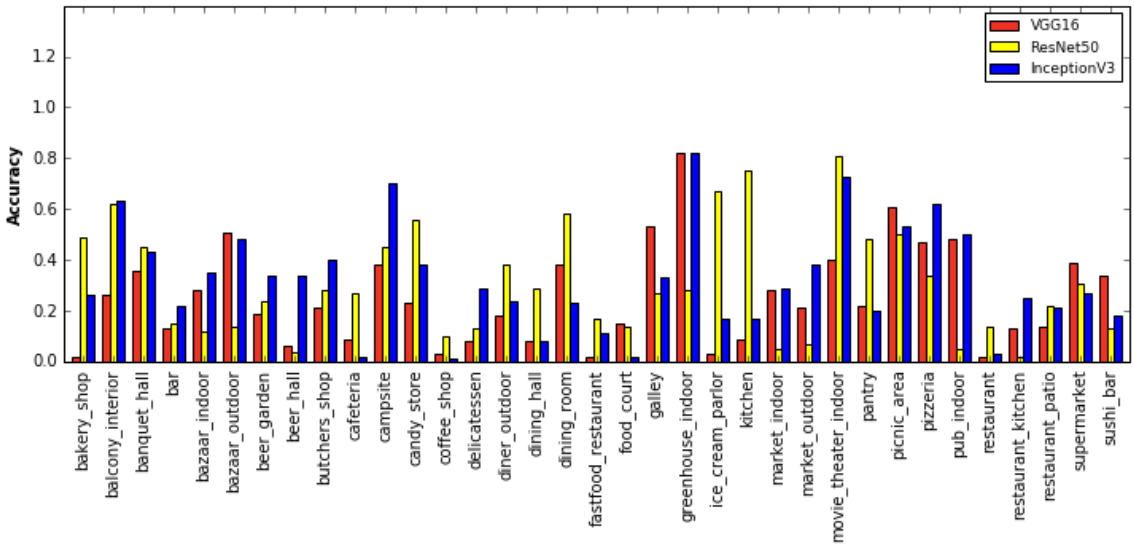


Figure 3.3: Classification accuracy per class (a) default network

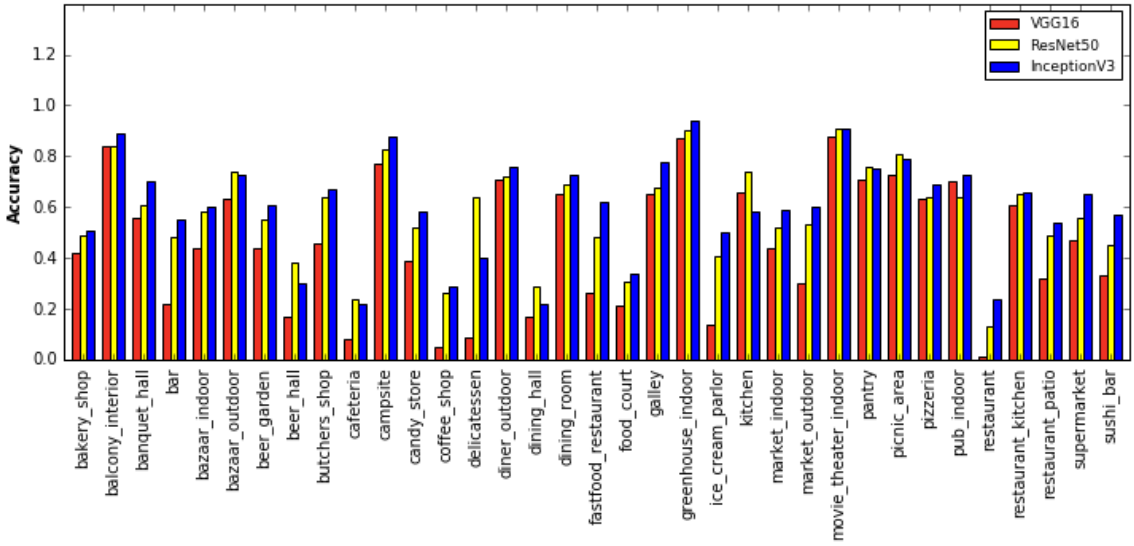


Figure 3.4: Classification accuracy per class(b) retrain last layer

VGG-16, ResNet-50 and Inception-V3. These models are trained and tested in three phases. Firstly, the default network with the number of parameters is fixed as the original network (Simonyan and Zisserman, 2014; He et al., 2016; Szegedy et al., 2016). Secondly, we retrain the last layer of the networks. Finally, we retrain the full networks and fine-tune their parameters for achieving a high classification rate. The figures show that retraining the full network architectures benefited a big development. The diagrams in Figure 3.3, 3.4 and 3.5 illustrate the overall

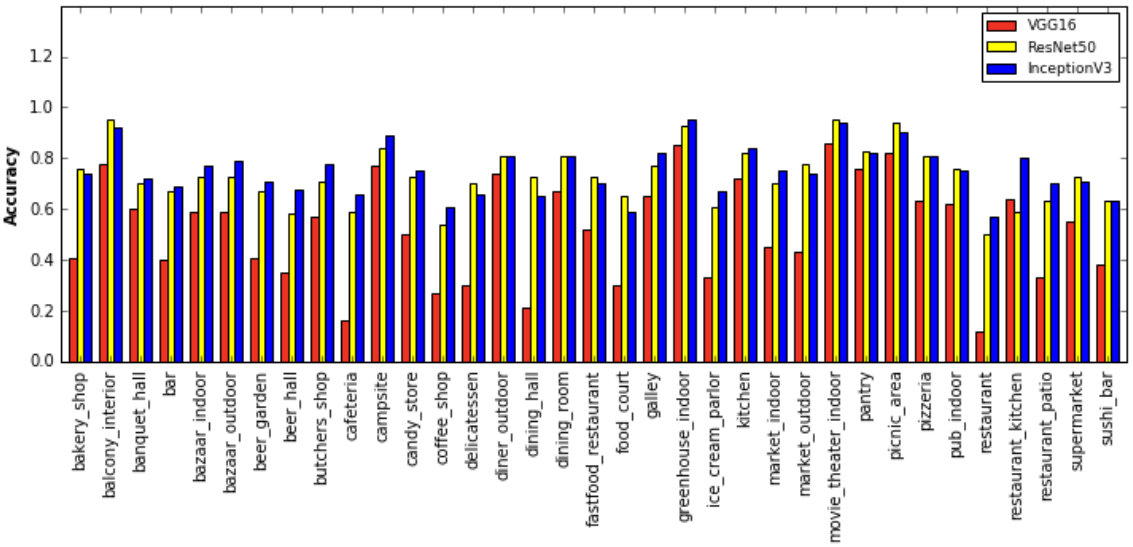


Figure 3.5: Classification accuracy per class (c) retrain full network

classification accuracy per class. With regards to the overall score, Figure 3.3 and 3.4 show that the default and the network with retrained last layer have not reached good accuracy for the fully retrained network shown in Figure 3.5. Furthermore, it also shows that VGG-16 has very poor accuracy in all three sections. Taking into account the accuracy per class for ResNet-50 and Inception-V3 by retraining the full network, some concrete results have been carried out. The fully retrained Inception-V3 model obtained very good accuracy on our dataset. After retraining the whole network, the prediction of some classes (e.g., balcony_interior, campsite, greenhouse_indoor and movie_theater_indoor) achieved more than 90% on the test dataset. We have the intuition that this is because the images from those classes have relatively low diversity. On the other hand, some classes (e.g., coffee_shop, food_court, restaurant and sushi_bar) have high diversity and overlapping with other classes, that leads to low accuracy. The Inception-V3 model performed very well in our overall classification procedure. However, ResNet-50 also had good achievements with few classes (e.g. supermarket) compared to the Inception-V3. The accuracy of the classification results for our models are listed in Table 3.3. Figure 3.6 shows the confusion matrix of our retrained Inception-V3 model for food-related scene classification. We show some miss-classified examples images in Figure 3.7. By analyzing the wrongly classified images, we found that the misclassifications are

Table 3.3: Comparison of accuracy among the models

Deep Feature	Accuracy (%)		
	Default Network	Network with Retrained Last Layer	Fully Retrained Network
VGG-16	25.14	45.74	52.22
ResNet-50	30.54	57.45	73.17
Inception-V3	32.02	60.34	**75.22**

usually occurring for a similar context of images in different classes (e.g., restaurant, restaurant_kitchen, restaurant_patio). Figure 3.8 shows examples of correctly classified images.

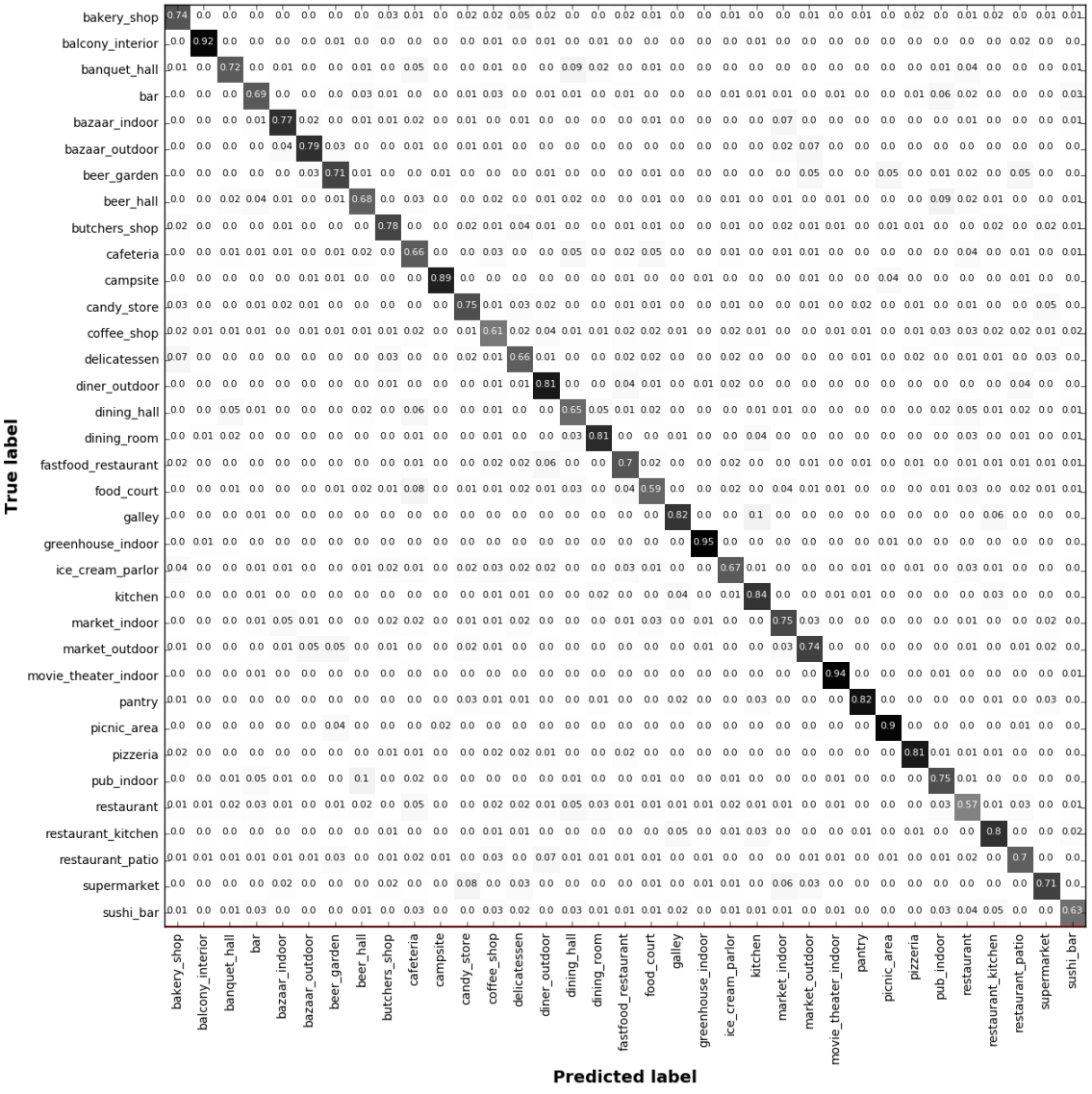


Figure 3.6: Confusion matrix of food-related scene classification.

Figure 3.7: Example images of some misclassified category

Figure 3.8: Examples of correctly classified food places images in "FoodPlaces" dataset

3.4 Conclusion and Future Work

In this chapter, we presented food places scene classification methods by using transfer learning of different convolutional neural network models. To achieve higher accuracy, we fine-tuned all the network layers that increased the classification performance. The obtained results imply that the Inception-V3 architecture can learn the features of food-related scene places of given image samples with a high classification rate of 75.22%. Note that the diversity of our dataset is very high because of mixing images from different datasets. The direction of our future research hints to continue with the fusion of the ResNet and the Inception model to improve the classification accuracy of food-related environment classification and analysis.

Chapter 4

Recognizing Food Places in Egocentric Photo-streams using Multi-scale Atrous Convolutional Networks and Self-Attention Mechanism

4.1 Introduction

Overweight and obesity yield many major risk factors for chronic diseases, including diabetes, cardiovascular diseases and cancer. According to the statistics given by World Health Organization (WHO) (WHO, 2018b), the obesity rate has nearly tripled since 1975. In 2016, more than 1.9 billion adults with age 18 years and older were counted overweight through the world, out of which 650 million were obese (Hales et al., 2018; Peralta et al., 2018). Comparing the death reason for people shows that overweight and obesity kill more people than underweight and malnutrition (Keys, 1980). Therefore, the concern of preventing obesity is highly demanding in developed countries. On the other hand, the cost of health services caused by overweight and obesity are increasing for the government every year to billions of dollars (Finkelstein et al., 2009). For example, the obesity medical cost in Europe was estimated at around €81 billion per year in 2012. In keeping with

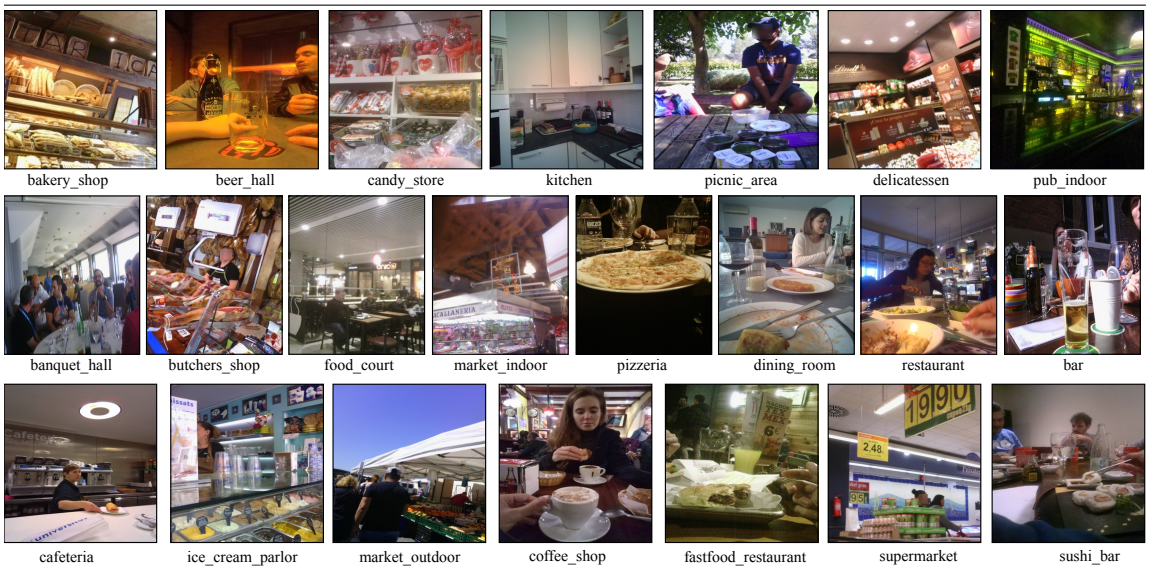


Figure 4.1: Examples of food places collected from the EgoFoodPlaces image dataset.

the WHO estimates on obesity expenditure, this was 2%-8% of the total national expenditure in the 53 European countries (Cuschieri and Mamo, 2016).

Food environment, adverse reactions to food, nutrition, and physical activity patterns are relevant aspects of the health care professional to consider when treating obesity. Recent studies have shown that 12 cancers are directly linked to overweight and obesity (Allen, 2018). The food that we eat, how active we are and how much we weigh have a direct influence on our health. Thus, by observing unhealthy diet patterns, we can create a healthy diet plan that can play a major role in our fight against obesity and being overweight. Therefore, diet patterns are important key factors that have to be analyzed for preventing overweight and obesity.

Conventional nutrition diaries are not good enough for tracking the lifestyle and food patterns properly since they need a huge amount of human interaction. Nowadays, mobiles phones are also used to keep track of one's diet by keeping a record of food intake and their respective calories. However, this is done by taking the photos of the dishes, which can make people uncomfortable (Redbook, 2017). For this reason, we need an automatic system that can correctly record the user food patterns and help to analyze the lifestyle and nutrition as well. To track the food patterns, we need to answer about three questions: where, how long and with whom

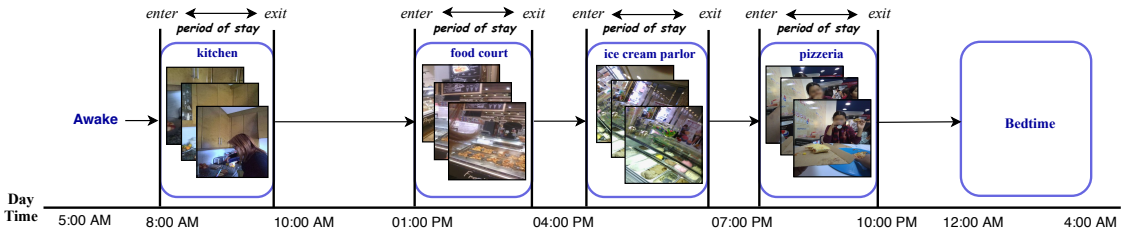


Figure 4.2: Examples of daily log that shows time spent in different food places.

the person is eating. These answers can discover the details of people nutritional habits, which can help to improve their healthy lifestyle and prevent overweight and obesity. In this work by analyzing daily user information captured by a wearable camera, we focus on the places or environment that users are commonly eating in, which is also called "food places".

Recording daily user information by the traditional camera is difficult. Therefore, we prefer to use wearable cameras, such as life-logging camera, being able to collect daily user information (see Figure 4.1). These cameras are capable of frequently and continuously capturing images that record visual information of our daily life known as "visual life-logging". It can collect a huge number of images by non-stop image collection capacity (1-4 per minute, 1k-3k (1k=1000) per day and 500k-1000k per year). These images can create a visual diary with activities of the person living with unprecedented details (Bolanos et al., 2017). The analysis of egocentric photo-streams (images) can improve the people lifestyle by analyzing social pattern characterization (Aghaei et al., 2018) and social interactions (Aghaei et al., 2015), as well as generating storytelling of first-person days (Bolanos et al., 2017). Besides, the analysis of these images can greatly affect human behaviours, habits, and even health (Grimm and Steinle, 2011). One of the personal tendencies of people is food events that can badly affect their health. For instance, some people get hungrier if they continuously see and smell food, consequently they end up eating more (Kemps et al., 2014; de Wijk et al., 2012). Also, it is well-known that people going to shop hungry, buy more and less healthy food. Thus, monitoring the duration of food intake and the time people spend in the food-related environment can help them get aware of their habits and improve their nutritional behaviour.

The motivation behind this research is two-fold. Firstly, using a wearable camera is to capture images related to food places, where the users are engaged within foods (see Figure 4.1). Consequently, these images of visual life-logging can give a unique opportunity to work on food pattern analysis from the individual's viewpoint. Secondly, the analysis of everyday information (entering, exiting and time of stay as shown in Figure 4.2) of visited food places can enable a novel healthcare approach that can help to manage better diseases related to nutrition, like obesity, diabetes, heart diseases, and cancer.

In this chapter, we propose two approaches to solve this problem: image-level and event-level food place recognition. In the image-level approach, food-related places of each individual in the scene with the camera-wearer is established in every image of the photo-stream. The presence of a food-related scene is decided in every single image separately and eventually, if the found scenes are related to the food places then store the information about the places, otherwise discard all non-relevant places information. In the event-level analysis, we make use of the temporal evolution of food-related scenes along with a potential event (period of stay). We aim to describe how the camera camera-wearer (first person) is engaged with the food places.

4.2 Related Works

Early work of places or scene recognition in conventional images has been discussed in the literature by applying classical approaches (Oliva and Torralba, 2002; Luo and Boutell, 2005; Cao and Fei-Fei, 2007; Yu et al., 2013). The traditional scene classification methods can be classified into two main categories: generative models and discriminative models. Generative models are generally hierarchical Bayesian systems to characterize a scene, which can represent different relations in a complex scene (Li et al., 2009; Qin and Yung, 2010; Sudderth et al., 2005). Discriminative models are to extract dense features of an image and encode the features into a fixed-length description to build a reasonable classifier for scene recognition (Elfiky et al., 2012; Li et al., 2010). The discriminative classifiers,

such as logistic regression, boosting and Support Vector Machine (SVM) was widely adopted for scene classification (Parizi et al., 2012). In (Lazebnik et al., 2006), the authors recognized 15 different categories of outdoor and indoor scenes by computing histograms of local features of image parts. In turn, Quattoni and Torralba (2009) proposed a scene classification method for indoor scenes (i.e., a total of 67 categories of scenes; 10 of them are related to food places). The method is based on a combination of local and global features of the input images.

Recently, the Convolutional Neural Networks (CNNs) have shown fruitful applications to digits recognition. CNN's have become a more powerful tool after introducing AlexNet (Krizhevsky et al., 2012) based on the large-scale dataset called "ImageNet" (Russakovsky et al., 2015a). Afterwards, the history of CNN evolution began with many breakthroughs, such as VGG-16 (Simonyan and Zisserman, 2014), Inception (Szegedy et al., 2015) and ResNet-50 (He et al., 2016). The era of places classification turned into new dimensions after introducing two large-scale places datasets, Places2 (Zhou et al., 2014) and SUN397 (Xiao et al., 2010) with millions of labelled images. The combination of using deep learning models with large-scale dataset outperforms the traditional scene classification methods (Zhou et al., 2018).

An overall of the state-of-the-art places or scene classification based on deep networks has been discussed in a review article presented in (Zhou et al., 2018). However, the performance of *scene recognition* challenges shown in (Zhou et al., 2018) has not achieved the same level of success as *object recognition* challenges (Russakovsky et al., 2015a). This outcome showed the difficulty of the general classification problem between scene and object level, as a result of large different places surroundings people (e.g., 400 places in Places2 dataset (Zhou et al., 2018)). In (Zheng et al., 2014), the authors proposed a probabilistic deep embedding framework for analyzing scenes by combining local and global features extracted by a CNN network. In addition, two separate networks called "Object-Scene CNN's" proposed in (Wu et al., 2015), in which a composed model of 'object net' and 'scene net' for aggregating information from the outlook of objects performs scene recognition. The two networks were pre-trained on the

ImageNet dataset (Russakovsky et al., 2015a) and Places2 dataset (Zhou et al., 2018), respectively. Indeed, many of deep architectures were evaluated on these datasets based on the conventional images. None of them is tested on the egocentric images that themselves represent a challenge for image analysis.

Recently, egocentric image analysis is a very promising field within computer vision for developing algorithms for understanding the first-person personalized scenes. Many classifiers were used to classify 10 different categories of scenes based on egocentric videos (Furnari et al., 2016). They trained the classifiers by using One-vs-All cross-validation. Moreover, a multi-class classifier with a negative-rejection technique was proposed in (Furnari et al., 2017). Both works (Furnari et al., 2016, 2017) considered only 10 categories of scenes, 2 of them are related to food places (i.e., *kitchen* and *coffee machine*). Moreover, some places related to food and type of food are classified in (SARKER et al., 2017; Sarker et al., 2018a) by using conventional images from the Places2 and CuisineNet dataset (Zhou et al., 2018; Sarker et al., 2018a).

In the image-level analysis work (Sarker et al., 2018c), we introduced a deep network named "MACNet" based on multi-scale atrous convolutional networks (Chen et al., 2018) for food places classification. The MACNet model is based on a pre-trained ResNet and works on images without using any time dependence (Sarker et al., 2018c). Besides, food places recognition is still a challenge due to the big variety of food places environments in real-world, and the wide range of possibilities of how a scene can be captured from the person's point of view. Therefore, we re-define our problem based on the relevant temporal intervals (period of stay time). This period is divided into a set of events that is a sequence of correlated egocentric photos. To classify the events in our event-level models (Sarker et al., 2019b), a self-attention deep model will then be used to classify these events. To the best of our knowledge, this is the first work on the food places pattern classification based on an event of a stream of egocentric images to create intelligent tools for food-related environment monitoring.

4.3 Methodology

We, humans, are naturally able to classify a place by simply looking at a sequence of images but it is a little bit hard to recognize a place by seeing a single image from them. Applying this concept to the computer program for building deep models that can able to achieve human-level performance. We split our task into two sections, image-level and event-level analysis.

4.3.1 Image-level Analysis

The proposed deep model for image-level analysis, MACNet, is based on multi-scale Atrous convolution networks for extracting the key patterns of food places in the input egocentric photo-streams. The multi-scale features are used to fine-tune four layers of a pre-trained ResNet-101 model as shown in Figure 4.3.

4.3.1.1 Network Architecture

The input images are scaled to five resolutions (i.e., the original size and four different resolutions) as shown in figure 4.3. The five images with different resolutions feed to Atrous convolution networks (Chen et al., 2018). In MACNet, five blocks of Atrous convolution network with three different rates per block are used to extract the key

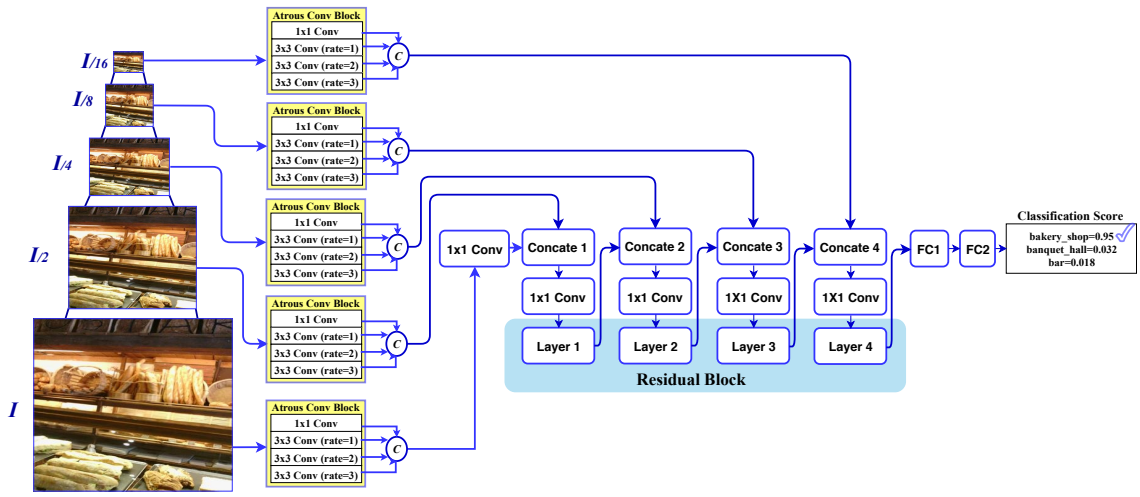


Figure 4.3: Architecture of proposed model (MACNet) for image-level analysis of food places classification.

features of an input image. Atrous convolution network allows us to explicitly extract features with different scales. In addition, it adjusts the filter's size with the rate value to capture multi-scale information, generalizes standard convolution operation. We used 3×3 kernels in all blocks with different rate values set to 1, 2 and 3. More details about these networks presented in (Chen et al., 2017b) and Chen et al. (2018).

Following, four pre-trained ResNet-101 blocks are then used to extract 256, 512, 1024 and 2048 feature maps, respectively as shown in figure 4.3. The four ResNet-101 layers are with stride 2.0. Thus, the final output size of the last ResNet block is 1/16 of the input image size. Indeed, each ResNet is corresponding to a resolution level in the image pyramid. Each output of the five Atrous network blocks is followed by a pointwise convolution (i.e., 1×1 convolution) to reduce the computation complexity and the number of channels to be compatible with the input channels accepted by the corresponding ResNet layer. All Atrous convolution networks and 1×1 convolution are randomly initialized. The output of the fourth ResNet layer feeds to a fully connected layer with 1024 neurons followed by another fully connected layer with 512 neurons. A dropout function with 0.5 is used for reducing overfitting in the two fully connected layers. A ReLU function is also used as an activation function for the first fully connected layer. In turn, a softmax function (i.e., normalized exponential function) is finally utilized as a logistic function for producing the final probability of the input image to each class. The two fully connected layers are randomly initialized.

4.3.2 Event-level Analysis

Recently, the Recurrent Neural Networks (RNNs) and attention-based models are widely used in the fields of Natural Language Processing (NLP), such as (Vaswani et al., 2017) for image captioning (Xu et al., 2015b), for video captioning (Hori et al., 2017), and sentiment analysis (Jabreel et al., 2017, 2018). In these approaches, a query vector is commonly used, which contains relevant information (i.e., in our case it is image-level features) for generating the next token to pick relevant parts of the input as supplementary context features. The attention models can be classified into two categories (Xu et al., 2015b), namely local (hard) and global (soft) attention. The

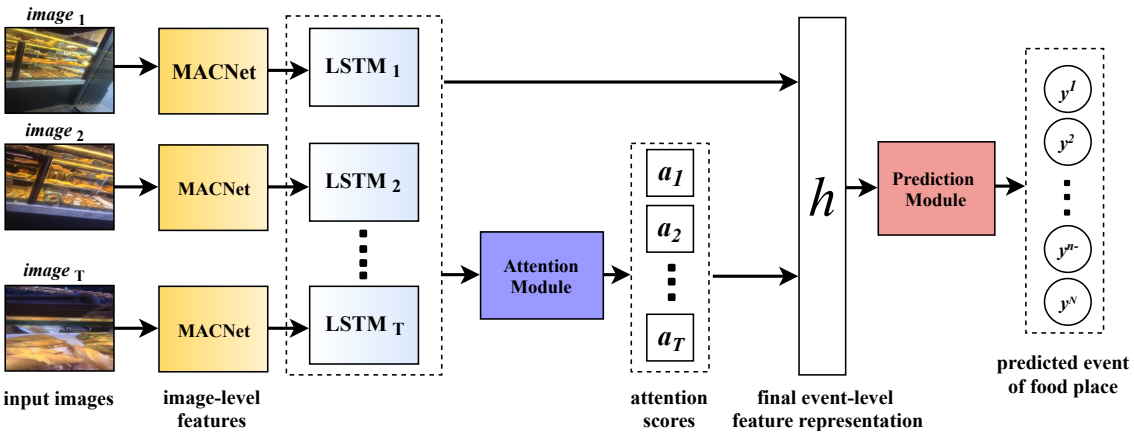


Figure 4.4: Architecture of our proposed attention-based model for event-level analysis of food places classification.

hard attention selects only a part of the input, which is non-differentiable that needs a more complex algorithm, such as variance reduction or reinforcement learning to train. In turn, soft attention is based on a softmax function to create a global decision on all parts of the input sequence. In addition, back-propagation is commonly used for training the attention models with both mechanisms in various tasks.

One of the effective soft-attention models is a self-attention mechanism (Lin et al., 2017c) with no extra queries. The self-attention mechanism can easily estimate the attention scores based on a self-representation. In this work, our attention model follows the self-attention scheme, where features extraction from the input images is done using the pre-trained MACNet model. LSTM cells are used to compute the attention scores. That is done by feeding these image-level features to an attention module to generate event-level features that the prediction module uses to classify the input event.

4.3.2.1 Network Architecture

The main framework of the proposed attention-based model for the event-level analysis of food places classification is illustrated in Figure 4.4. The proposed model consists of three major modules: features extraction, attention and prediction modules. The feature extraction module is based on the MACNet (Sarker et al., 2018c) model that is fed by input image from a food place event, see Figure 4.3. We

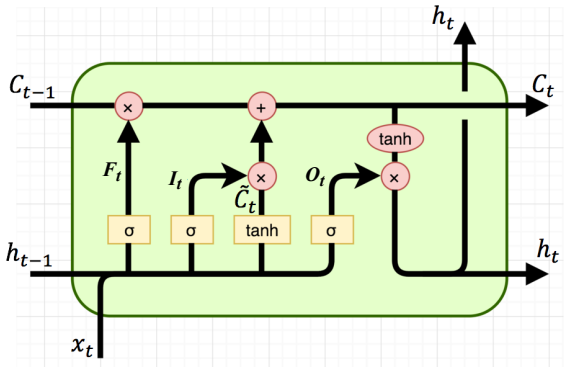


Figure 4.5: Standard architecture of an LSTM cell.

feed a sequence of egocentric photo-streams (an event) as an input to the MACNet and extracted the features from them. We used the same architecture for feature extractor until Residual block (layer 4) without two FC layers of MACNet shows in Figure 4.3, which was utilized for classification.

In the second step, a Long Short-Term Memory (LSTM) unit (LSTM cell) (Hochreiter and Schmidhuber, 1997) is applied designed to learn long-term dependencies features of all images per event. This unit consists of a number of LSTM cells. Figures 4.5 illustrates the LSTM cells properties. A classical LSTM cell consists of three sigmoid layers: a forget gate layer, an input gate layer, and an output gate layer. The three layers determine the information to flow-in and flow-out at the current time step. The mathematical definitions of these layers can be defined as:

$$F_t = \sigma(W_F.[h_{t-1}, x_t] + b_F), \tag{4.1}$$

$$I_t = \sigma(W_I.[h_{t-1}, x_t] + b_I), \tag{4.2}$$

$$O_t = \sigma(W_O.[h_{t-1}, x_t] + b_O), \tag{4.3}$$

where, σ represents the sigmoid function, x_t is the input features vector at time t, h_{t-1} is the output state of the LSTM cell at the previous step at time $t-1$, F_t, I_t, and O_t are the outputs of the three gates layers at time t, W_j, and b_j are a weight matrix and a bias scalar for a layer, where j is for F, I or O layers. For updating the cell state, the LSTM cell also needs a *tanh* layer to create a vector of new candidate values, $\tilde{C}_t$, which can be computed after the information coming from the input gate

layer by:

$$\tilde{C}_t = tanh(W_C.[h_{t-1}, x_t] + b_C], \tag{4.4}$$

where W_C and b_C are a weight matrix and a bias scalar for the $tanh$ layer. The old cell state, C_{t-1}, to the new cell state, C_t can be updated by combining the outputs of the forget and the input gate layers by:

$$C_t = F_t * C_{t-1} + I_t * \tilde{C}_t \tag{4.5}$$

Finally, the output state of the LSTM cell is:

$$h_t = O_t * tanh(C_t). \tag{4.6}$$

In the model, the outputs of the MACNet model are the features extracted from the input images of an event x_0, x_1, ..., x_T. These features are fed to a set of LSTM cells, for capturing additional context dependencies features. Assume we have T number of LSTM cells, $\{LSTM_1, ..., LSTM_T\}$, $LSTM_t \in \mathbb{R}^H$, where T is the number of images and H is the dimension of the extracted features vector. The output features of the LSTM cells are sequentially fed to an attention module to ensure that the network can increase its sensitivity to the important features, and suppress less useful features. The attention module will be learned how to average image-level features in a weighted manner. The weighted average is obtained by weighting each image-level features by a factor of its product with a global attention vector. The features vector of each image and the global attention vector will be trained and learned simultaneously using a standard back-propagation algorithm. In our proposed model, we use the dot product between global attention vector V and image-level feature $LSTM_t$ as a score of the t-th image. Thus this score can be computed as:

$$S_t =< V, LSTM_t > . \tag{4.7}$$

The global attention vector, $V \in \mathbb{R}^H$ is initialized randomly and learned simultaneously by the network. To construct image-level features for different

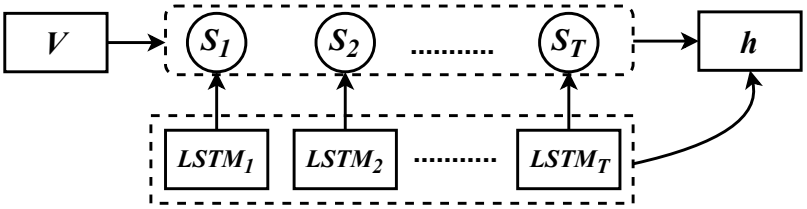


Figure 4.6: Global self-attention mechanism for final event-level feature representation.

food-places events, the global attention vector, V can learn the general pattern of the event relevance of images. The architecture of the global self-attention mechanism is shown in Figure 4.6. Multiple information of successive images is aggregated into a single event-level vector representation with attention. The attention mechanism computes a weighted average over the combined image-level features vectors, and its main job is to compute a scalar weight to each of them. For constructing the final event-level representation, it is also not differentiated whether the images belong to the target event or any other events.The attention module measures a score, S_t for each image-level features $LSTM_t$ and normalizes it by a softmax function as follows:

$$\alpha_t = \frac{exp(S_t)}{\sum_{t=1}^{T} exp(S_t)}, \tag{4.8}$$

where α is the probabilistic heat-map. Thus, the image-level features $LSTM_t \in \mathbb{R}^H$ are then biased by the corresponding attention scores. The final event-level features, h are the element-wise weighted average of all the image-level features defined as:

$$h = \sum_{t=1}^{T} \alpha_t LSTM_t, \tag{4.9}$$

where h is the event-level features that will be used to automatically train the prediction module to predict the events of a period of stay in a food-place. There are various type of the prediction modules available in the literature. In this work, a fully connected neural network is used as a multi-label event prediction module:

$$\hat{y}^n = p(y^n|h) = \frac{1}{1 + e^{-}(w^n h + b^n)} \in [0, 1], \tag{4.10}$$

where $\hat{y}^n$ is the predicted label, y^n is the ground-truth of the n-th event, $n = 1$ to N,

N is the total number of events samples, and w^n and b^n are the classification weight and biasing parameters, respectively, for predicting the n-th event. The whole model trained end-to-end by minimizing the multi-label classification loss is given by:

$$\ell = -\frac{1}{N}\sum_{n=1}^{N} E(y^n, \hat{y^n}), \tag{4.11}$$

where E is the cross-entropy function.

4.4 Experimental Results

4.4.1 EgoFoodPlaces dataset

In this work, we introduced a new egocentric dataset "EgoFoodPlaces" for food places classification. Our egocentric dataset, "EgoFoodPlaces", was constructed by 16 users using a lifelogging camera (i.e., narrative clip 2 (Narrative, 2017), which has an image resolution of 720p and 1080p by a 8-megapixel camera with an 86-degree field of view and capable of record about $4,000$ photos or 80 minutes of 1080p video at 30fps. Figure 4.1 shows some example images from the "EgoFoodPlaces" dataset. The user fixed the camera to his/her chest from morning to night before sleeping for capturing the visual information about his/her daily environment. Thus, sets of egocentric photo-streams (events) exploring the users daily food patterns (e.g. a person spends a specific time in a food-place, such as a restaurant, cafeteria, coffee shop, etc.) were captured, see Figure 4.2. Every frame of a photo-stream is recording first-person personalized scenes that will be used for analyzing different patterns of the user lifestyle.

However, in "EgoFoodPlaces", the captured images have different challenges, such as the blurriness (the effect of the user' motion), black, ambiguous and occluded images (occluded by the user hand or other body parts) during the streaming, which is not good for the entire system. All these challenges reduce the accuracy rate of a recognition system. Therefore, some pre-processing techniques are necessary to be applied to refine the collected images.

For removing the blurry images, we compute the blurriness amount in each image using the variance of the Laplacian. The blurriness amount is calculated by a pre-defined threshold (i.e. in this work, the threshold value is set to 500). If the variance is lower than the threshold, then the image is considered blurry. Particularly, if the image contains high variance, the image has a widespread response of both edge-like and non-edge indicating to an in-focus image. In turn, if the variance is low, the image has a tiny spread of responses specifying that the number of edges appearances in the image is very small and the image is blurred. In turn, *for removing the black, ambiguous and occluded images* from our dataset, the K-Means clustering algorithm was used with $K = 3$ (i.e., red, green and blue). If 90% of the pixels of an image are clustered to a dominant color, we consider that the image is not informative enough, and it is eliminated from the dataset.

Moreover, the "EgoFoodPlaces" dataset has some unbalanced classes. However, it is not possible to make it a balanced dataset by reducing images from other classes, since some classes have a very small number of images. The classes with few images are usually related to some food places that the users do not spend much time at them (e.g. butchers shop). In turn, some classes of a big number of images are related to places with rich visual information that refer to daily contexts (e.g. kitchen, supermarket), or places, where people spend more time (e.g. restaurant). We labelled our dataset by taking the reference classes names related to food scenes of the public Places2 dataset (Zhou et al., 2018). Initially, we chose 22 common food-related places that people often visited for our dataset. The food-related places that user visited very rarely (e.g. beer garden), were excluded from our dataset.

Finally, the 16 users recorded their period of stay (the exact time) in any food place visited during capturing the photo-streams. Afterwards, we created the events of each class by selecting the maximum correlated frames from that period. The period of stay is divided into a set of events. Each event is around 10 seconds. We select 10 seconds because we need to keep the similarity between the consequent frames. Since our wearable camera is adjusted to capture one frame per second, one event will contain 10 consequent frames. For instance, assume a user visited a bar for

Table 4.1: The distribution of images per class in the EgoFoodPlaces dataset.

	Train		Val		Test		Total	
Classes	images	events	images	events	images	events	images	events
bakery shop	356	36	108	11	128	13	592	60
banquet hall	420	42	150	15	146	15	716	72
bar	1320	132	410	41	730	73	2460	246
beer hall	600	60	110	11	344	35	1054	106
butchers shop	261	27	60	6	50	5	371	38
cafeteria	1443	145	200	20	370	37	2013	202
candy store	360	36	80	8	90	9	530	53
coffee shop	2060	206	260	26	590	59	2910	291
delicatessen	680	68	80	8	50	5	810	81
dining room	3020	302	420	42	930	93	4370	437
fastfood restaurant	920	92	150	15	330	33	1400	140
food court	200	20	90	9	40	4	330	33
ice cream parlor	160	16	50	5	60	6	270	27
kitchen	3300	330	400	40	990	99	4690	469
market indoor	800	80	150	15	210	21	1160	116
market outdoor	1313	132	60	6	250	25	1623	163
picnic area	667	67	140	14	260	26	1067	107
pizzeria	1120	112	370	37	600	60	2090	209
pub indoor	372	38	60	6	150	15	582	59
restaurant	4551	456	550	55	1120	112	6222	623
supermarket	3812	382	862	87	1423	143	6097	612
sushi bar	1270	127	340	34	426	43	2036	204
Total	29005	2909	5100	511	9287	932	**43392**	**4352**

10 minutes. Thus, for a minute, we will have 6 events (60 seconds/10 seconds) and 60 events for the whole 10 minutes. The 22 classes of food places in "EgoFoodPlaces" are illustrated in Table 4.1.

For the training, the dataset was split into three subsets: train (70%), validation (10%) and test (20%). The images of each set were not randomly chosen to avoid taking similar images from the same events. Thus, we split the dataset based on event information to make the dataset more robust to train and validate the models.

4.4.2 Experimental Setup

The proposed models were implemented in PytorchPaszke et al. (2017): an open source deep learning library. The Adam (Kingma and Ba, 2014) algorithm is used for model optimization. The "step" learning rate policy (Sebag et al., 2017) is used with

the base learning rate of 0.001 with 20 as a step value. For the LSTM cells, we used a hidden layer size of 2048 that is similar to the output size of the MACNet feature. The number of layers is 6 and the dropout rate is 0.3. In turn, for self-attention, 22 layers are used for getting the attention score of 22 classes (number of classes in "EgoFoodPlaces"). Besides, data augmentation is applied for increasing the dataset size and variation. We performed the random crop, image brightness and contrast change with 0.2 and 0.1, respectively. We also use image translation of 0.5, a random scale between 0.5 and 1.0, and random rotation of 10 degrees. The batch size is set to 64 for training with 100 epochs. The experiments are executed on NVIDIA GTX1080-Ti with 11 GB memory taking around one day to train the network. All the above parameters are used for testing the model as well.

4.4.3 Evaluation

To evaluate the proposed MACNet and MACNet+SA models quantitatively, we compared it with the state-of-the-art in terms of the average F_1 score, and the classification accuracy rate.

The F_1 score can be defined as:

$$F_1 \text{ score} = 2 \times \frac{Precision.Recall}{Precision + Recall}, \tag{4.12}$$

where precision is the number of true positives divided by the total numbers of actual results, and computed as:

$$Precision = \frac{True\ positive}{True\ positive + False\ positive}, \tag{4.13}$$

In turn, recall is the number of true positives divided by the total number of predicted results by the classifier, and computed as:

$$Recall = \frac{True\ positive}{True\ positive + False\ negative}. \tag{4.14}$$

Table 4.2: Average F_1 score of VGG-16 (Simonyan and Zisserman, 2014), ResNet-50 (He et al., 2016), Inception-V3 (Szegedy et al., 2016), the proposed MACNet (Sarker et al., 2018c) and the proposed MACNet+SA (Sarker et al., 2019b) model using both validation and test sets from EgoFoodPlaces dataset.

Categories	VGG-16		ResNet-50		Inception-V3		MACNet		MACNet+SA	
dataset	val	test	val	test	val	tests	val	test	val	test
bakery shop	0.77	0.59	0.72	0.65	0.77	0.75	0.74	0.68	**0.85**	**0.84**
banquet hal	0.71	0.48	0.62	0.38	0.73	0.50	0.64	0.51	**0.75**	**0.70**
bar	0.66	0.52	0.37	0.36	0.74	0.56	0.65	0.58	**0.85**	**0.73**
beer hall	0.77	0.48	0.92	0.45	0.91	0.40	0.95	**0.51**	**0.96**	0.44
butchers shop	0.71	0.83	0.72	0.91	0.72	0.89	**0.79**	**0.92**	0.73	0.88
cafeteria	0.61	0.47	0.64	0.60	0.70	0.59	0.78	0.63	**0.94**	**0.78**
candy store	0.65	0.59	**0.66**	0.63	0.65	0.57	0.63	**0.64**	0.64	0.58
coffee shop	0.45	0.71	0.57	0.71	0.66	0.68	0.89	0.75	**0.93**	**0.87**
delicatessen	0.52	0.62	0.55	0.73	0.50	0.64	**0.69**	0.56	0.59	**0.75**
dining room	0.62	0.67	0.71	0.74	0.73	0.75	**0.92**	**0.87**	0.87	0.86
fastfood restaurant	0.33	0.44	0.33	0.49	0.32	0.50	**0.77**	0.56	0.68	**0.63**
food court	0.64	0.66	0.63	0.69	0.70	0.63	0.82	0.63	**0.86**	**0.73**
ice cream parlor	0.65	0.64	0.64	0.60	**0.72**	**0.69**	0.66	0.64	0.67	0.65
kitchen	0.79	0.85	0.91	0.89	0.88	0.87	0.90	0.89	**0.93**	**0.92**
market indoor	0.53	0.44	0.56	0.60	0.40	0.48	**0.81**	0.64	0.76	**0.82**
market outdoor	0.42	0.53	0.37	0.77	0.39	0.70	**0.61**	0.69	0.48	**0.78**
picnic area	0.51	0.44	0.59	0.47	0.49	0.45	0.68	0.46	**0.80**	**0.67**
pizzeria	0.77	0.62	0.39	0.48	0.81	0.67	0.68	0.67	**0.99**	**0.95**
pub indoor	0.86	0.49	**0.96**	0.88	0.93	0.70	0.95	**0.92**	0.94	0.83
restaurant	0.51	0.47	0.62	0.46	0.60	0.51	0.72	0.55	**0.85**	**0.66**
supermarket	0.80	0.81	0.81	0.86	0.83	0.84	0.71	0.88	**0.91**	**0.89**
sushi bar	0.78	0.44	0.88	0.44	0.76	0.43	0.95	0.73	**0**.99	**0.88**
Avg. F_1 score	0.66	0.62	0.68	0.65	0.72	0.66	0.79	0.72	**0.86**	**0.80**

4.4.4 Results and Discussions

In this section, we have compared the proposed MACNet (image-level) and MACNet+SA (event-level) model with four baseline methods: three common classification methods, VGG-16 (Simonyan and Zisserman, 2014), ResNet-50 (He et al., 2016), and Inception-V3 (Szegedy et al., 2016) for both validation and test sets.

Table 2 shows the average F_1 score of the event-level analysis model, MACNet+SA, and the four tested methods with the 22 classes of "EgoFoodPlaces". As shown, MACNet+SA yielded the highest average F_1 score of 0.86 and 0.80 for both validation and test sets, respectively. In addition, MACNet+SA achieved the highest F_1 score with the majority of classes in the two sets. In turn, the image-level

analysis method, MACNet provided acceptable average F_1 score of 0.79 and 0.73 with the validation and test sets, respectively, which is higher than the other three methods, VGG-16, ResNet-50 and Inception-V3. The Inception-V3 achieved average F_1 score comparable with MACNet with 0.72, and 0.66 on the two sets. In turn, ResNet-50 and VGG-16 yielded similar average F_1 score of about 0.65.

For the validation set, with 13 out of 22 classes, MACNet+SA yielded the highest F_1 score. In turn, with 6 out of 9 remaining classes, the predecessor MACNet achieved the highest F_1 score. While for candy store and pub indoor classes ResNet-50 had the highest F_1 score. For the ice cream parlour class, Inception-V3 model yielded the highest F_1 score. In turn, VGG-16 achieved the lowest F_1 score among the five tested methods for all classes.

For the test set, the MACNet+SA model yielded the highest F_1 score with 16 out of 22 classes. In turn, the MACNet model achieved the highest F_1 score in 5 out of 6 remaining classes. In turn, the Inception-V3 yielded the highest F_1 score for the ice cream parlour class. Moreover, both VGG-16 and ResNet-50 models achieved lower F_1 score than the rest of the tested models for all classes.

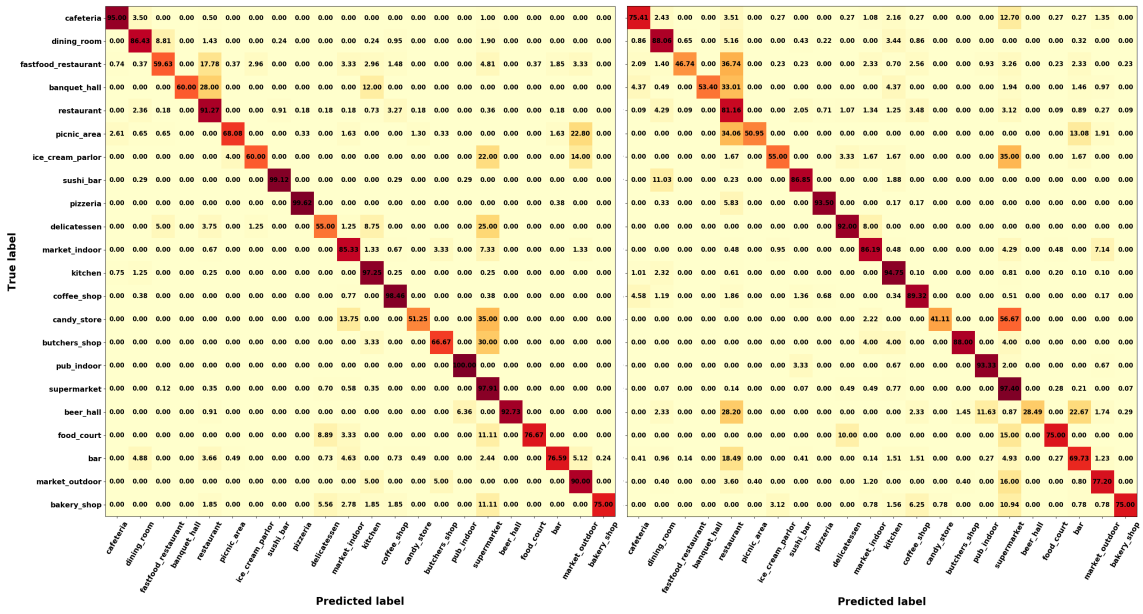
The MACNet+SA model yielded an average improvement of 7% and 8% in terms of the average F_1 score with the validation and test sets, respectively in a comparison of the second best state-of-the-art *i.e.*, MACNet. In some places like bar, cafeteria, picnic area, pizzeria and other places that need a sequence of images to describe them, MACNet+SA yielded a significant improvement of more than 10%. However, with some classes, such as butchers shop, dining room, market indoor and market outdoor, MACNet provided higher results than MACNet+SA showing that these type of places might not need to describe them with a sequence of images, and still images are able to describe these places.

In turn, Table 3 shows a comparison between the MACNet and MACNet+SA model with VGG-16, ResNet-50 and Inception-V3 in terms of Top-1 and Top-5 classification accuracy rates on both validation and test sets. It shows that MACNet+SA achieved the highest Top-1 and Top-5 accuracy rates with the two sets. Regarding the validation set, MACNet+SA yielded an improvement of 7% and

Table 4.3: Average Top-1 and Top-5 classifcation accuracy of VGG-16 (Simonyan and Zisserman, 2014), ResNet-50 (He et al., 2016), Inception-V3 (Szegedy et al., 2016), MACNet (Sarker et al., 2018c) and the MACNet+SA model using both validation and test sets from EgoFoodPlaces dataset.

Models	Validation		Test	
	Top-1	Top-5	Top-1	Top-5
VGG-16	0.66	0.87	0.62	0.86
ResNet-50	0.68	0.91	0.65	0.90
Inception-V3	0.72	0.91	0.66	0.88
MACNet	0.79	0.90	0.72	0.89
MACNet+SA	**0.86**	**0.93**	**0.80**	**0.92**

2% in terms of top-1 and top-5 rates, respectively, higher than the MACNet model achieving the highest classification rate among the four test models. In turn, for the test set, MACNet+SA yielded an improvement of 8% and 2% with top-1 and top-5 rates, respectively.



(a) **(b)**

Figure 4.7: The confusion matrices of (a) validation and (b) test sets of the EgoFoodPlaces dataset for evaluating our propose model.

Furthermore, Figure 4.7 shows a confusion matrix of the 22 classes of the EgoFoodPlaces dataset with the validation and test sets. The confusion matrix in Figure 4.7-(a) shows that the MACNet+SA model, with the validation set, was able to correctly classify the food-places events in most of the classes. However, it misclassifies events from a class to another. For example, MACNet+SA misclassifies

17.78% of fast food restaurant events to the restaurant class, in addition, 22.80% of picnic area events are misclassified with the outdoor market class, and 22% and 14% of ice cream parlour samples are misclassified with the supermarket and outdoor market classes, respectively. The confusion matrix also shows that 25% of delicatessen events are misclassified with the supermarket class, 35% of candy store samples are misclassified with the supermarket class, 28% of banquet hall samples are misclassified with the restaurant class, and 30% of butcher shop events are misclassified with the supermarket class. The confusion matrix in Figure 4.7 (b) shows that the MACNet+SA classification model with the test set misclassifies events from classes to restaurant, supermarket and bar classes. It shows 36.74%, 33.01%, 33.01%, 34.06%, and 18.49% of the events of the fast-food restaurant, banquet hall, picnic area, beer hall and bar classes are misclassified to the restaurant class. In addition, the confusion matrix shows 12.70%, 35%, 56.57%, 15%, 16%, and 10.94% of cafeteria, ice-cream parlour, candy store, food court, market outdoor and bakery shop events are misclassified with the supermarket class. Similarly, 13.08%, and 22.67% of the picnic area and beer hall events, respectively, are misclassified with the bar class. However, for all of these misclassifications events, there is a lot of similarity between their scenes in terms of the context and objects. Even, humans prone to weakly recognize such places many times.

Figure 4.8 shows examples of correct and incorrect predictions by the MACNet+SA model with the "EgoFoodPlaces" dataset. The first, third, fifth and seventh rows show that the MACNet+SA model can properly predict all images of events of the dining room, restaurant, sushi bar and banquet hall classes, respectively. In turn, second, fourth and sixth and last rows show incorrect predictions examples, in which one image or more of the dining room, restaurant, sushi bar and banquet hall events are misclassified. In the second row, images in first, second, third and fourth columns are correctly classified as a dining room class; whereas, the images in the last image is misclassified as a fast-food restaurant. In the fourth row, the restaurant class is correctly predicted with first and third images, while the second and last images are misclassified as a coffee shop and the dining room, respectively.

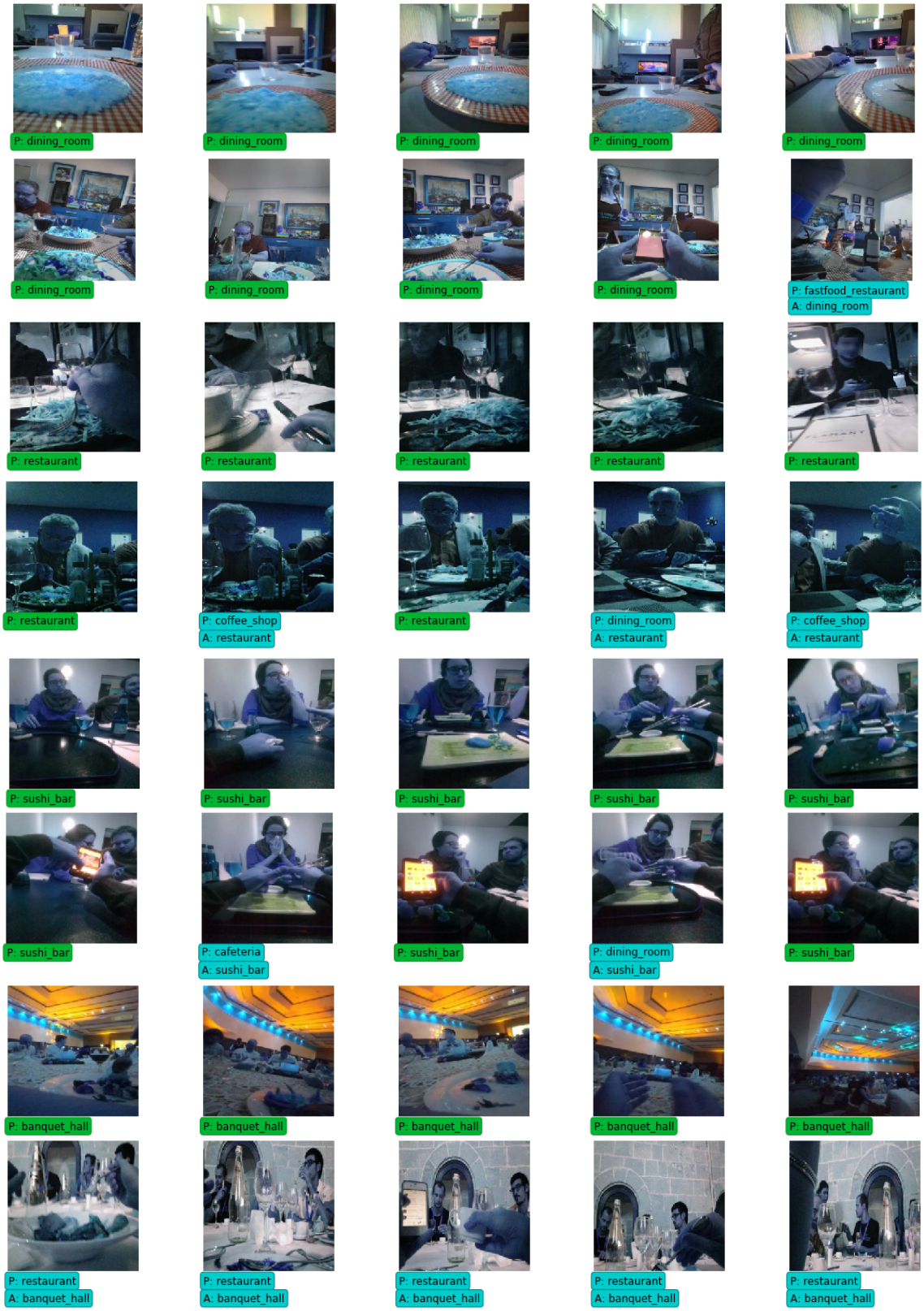


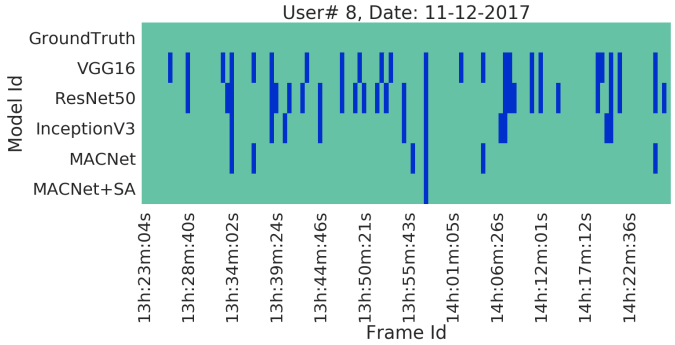
Figure 4.8: Examples of correct and incorrect predictions of MACNet+SA model with the input event (a sequence of images) of the validation set.

However, the Top-2 prediction is the correct class, restaurant. In the sixth row, the sushi bar class is correctly predicted with the first, third and last images. In turn, the second and fourth images are misclassified as cafeteria and dining room classes, respectively. In the last row, all images of a banquet hall event are predicted as a restaurant class. However, with all images, the Top-2 prediction is the banquet hall class.

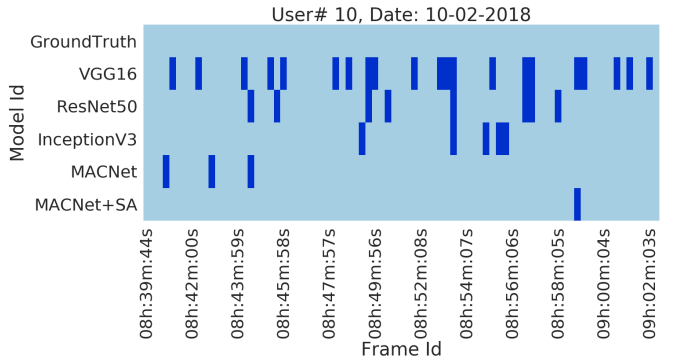
Figure 4.9: Examples of the resulting predictions (from Top-1 to Top-5) of the MACNet+SA model using validation dataset, where GT is the ground-truth label of the predicted class.

Figure 4.9 shows examples of predicted Top-1 to Top-5 accuracy. The first row shows that cafeteria, kitchen and restaurant images are properly classified with Top-1 classification accuracy rates of 93.06%, 84.99% and 89.67%, respectively. In turn, the second row shows the MACNet+SA model wrongly predicted restaurant, dining room and fast-food restaurant classes with the Top-1 accuracy. However, these classes barely appeared in Top-5 accuracy with a restaurant in Top-2, dining room in Top-3 and fast-food restaurant in Top-5, with a classification accuracy of 39.60%, 3.65% and 11.36%, respectively.

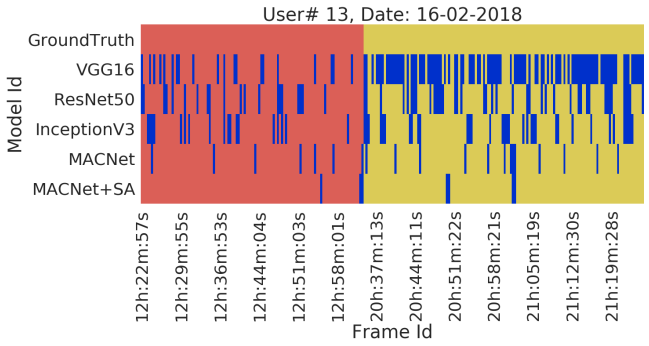
Figure 4.10 shows four periods of stays in six food places captured by four different users (users 8, 10, 13 and 16 of the "EgoFoodPlaces" dataset) in four different days. The user 8 visited a coffee shop for 59 minutes, and user 10 visited the bakery shop for 22 minutes. In addition, the third and fourth users visited two different food places: food court and sushi bar for user 13, whereas the kitchen and dining room for user 16. All events during each period were tested with the proposed



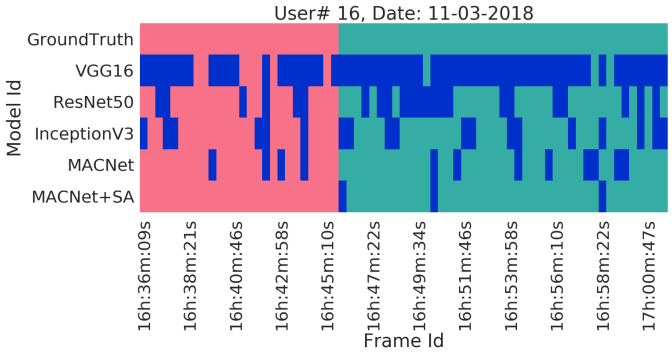
(a)



(b)



(c)



(d)

Figure 4.10: Resulted food places classification with four periods of stay in six food places (coffee shop, bakery shop, food court, sushi bar, kitchen and dining room) captured by four different users (users 8, 10, 13 and 16 of the EgoFoodPlaces dataset) in four different days from the validation set.

MACNet+SA model. For instance, for user 8, he spent 59 min in a coffee shop, we divided it into 354 events. In turn, for user 16, 54 events were included in his first stay in kitchen (i.e. 9 minutes), and 72 events during his stay inside a dining room (i.e. 12 minutes). One can notice that the proposed MACNet+SA model yielded the lowest misclassification rates in the four sequences of events. With the sequences of the events of user 8 and 10 in coffee and bakery shops, respectively, the MACNet+SA model misclassified only one event per every sequence. In the sequence of the third events of user 13, MACNet+SA misclassified two events in the food court and five events in the sushi bar. In turn, for user 16, the MACNet+SA model properly predicted all events of the kitchen. However, it misclassified three events in the dining room. Supporting the aforementioned results, the MACNet model provides the second rank after the MACNet+SA with misclassification of 6, 3, 19, and 12 events with user 8, 10, 13 and 16, respectively. In turn, the VGG-16 provided the worst classification rate among the all tested models. When considering capturing images of the daily life of persons and their environment, wearable devices with first-person cameras can raise some privacy concerns, since they can capture extremely private moments and sensitive information of the user. There are five steps of data privacy consideration in life-logging (Gurrin et al., 2014): capture, storage, processing, access and publication. The first three phases have no human involvement. In the final two stages, the data can be accessed by humans. To deal with the privacy issues in real-life applications, the images can be online processed with the trained model with only storing the logging information without any confidential data and avoiding to store the images during the logging process. Also, the user can handle the system with mobile apps to turn off in private moments and turn on when entering food places. Taking this viewpoint, we consider that the right to privacy in terms of life-logging refers to *the right to choose the composition and the usage of your life-log and the right to choose what happens to your representation in the life-logs of others* (Gurrin et al., 2014)

4.5 Conclusions

In this chapter, we designed two deep models for food places classification system, MACNet and MACNet+SA, for egocentric photo-streams captured during a day. The main purpose of this classification system is to later generate a dietary report to analyze people's food intake and help them control their unhealthy dietary habits. The proposed deep models are based on multi-scale atrous convolutional networks and a self-attention model with it. The proposed MACNet model used multi-scale atrous convolutional networks to classify still images. However, the proposed MACNet+SA using attention mechanism to classifies a sequence of images (called events) to get relevant temporal information about the food places. Image-level features are extracted by the MACNet model. The LSTM cells with a self-attention mechanism merge the temporal information of the sequence of the input images. The quantitative and qualitative results show that the proposed MACNet+SA model can outperform state of the art classification methods, as VGG-16, ResNet-50, Inception-V3 and MACNet. MACNet+SA on the dataset, EgoFoodPlaces, yields an average F_1 score of 86% and 80% on validation and test set, respectively. In addition, it yields a Top-1 accuracy of 86% and 80%, and a Top-5 accuracy of 93% and 92% on the validation and test sets, respectively. Future work aims at developing a mobile application based on the MACNet+SA model that integrates an egocentric camera with a personal mobile device to create a dietary report to keep a track on our eating behaviour or routine for following a healthy diet.

Chapter 5

CuisineNet: Food Attributes Classification using Multi-scale Convolution Network

5.1 Introduction

Food has different attributes, such as cuisine, course, nutrition, ingredients and flavours. The diversity of food has a strong effect on our social and personal life (Rozin et al., 1999). The cuisine is a particular procedure for preparing food-related to geographic locations. It plays a very important role in culture, which reflects its unique history, lifestyle, values, and beliefs, as well as people tend to identify themselves with their food. It also helps to easily understand people humerus. Finding the attributes of food from its images is a key role in different applications, such as studying food culture and preference, calorie approximation from food images and individualized recipe recommendation. The increase in on-line food-attributes sharing websites has provided rich data for food-related research. These websites generally have multi-modalities and multi-attributes. For instance, the well-known *Yummly* website (Yummly, 2009) is used for food-attributes with more than one million attributes of a large amount of metadata information. Some examples of the *Yummly*' attributes are shown in Figure 5.1. Every food item consists of a food image, textual information (i.e., name, ingredients and nutritions) and attributes (i.e., cuisine, course and flavours). In the literature, many works have

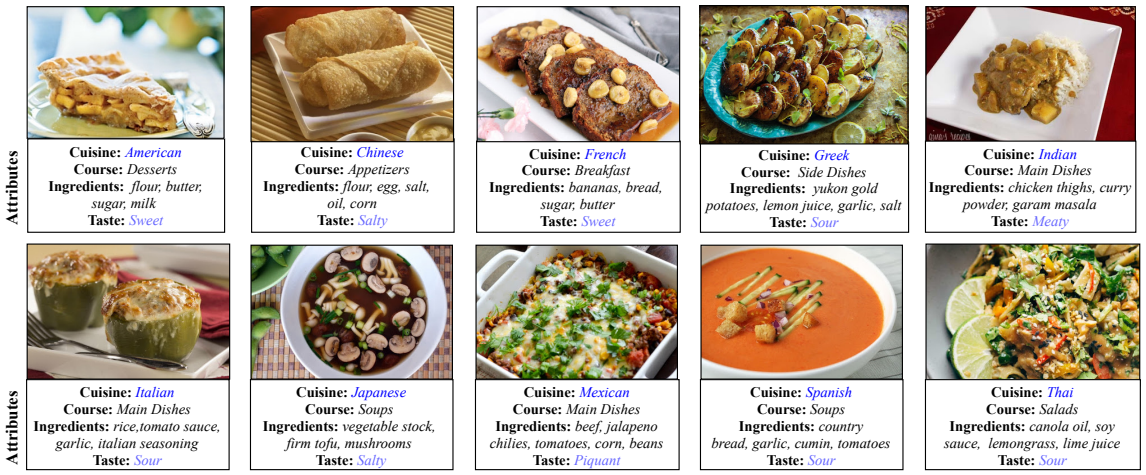


Figure 5.1: Some examples of food with their attributes from *Yummly*.

been proposed for food image recognition (Bossard et al., 2014), (Farinella et al., 2014). After the breakthrough of convolutional neural networks (CNN), other works have recently been developed for food classifications (Bolanos and Radeva, 2016), (Aguilar et al., 2017), food places recognition (SARKER et al., 2017). In addition, restaurant-specific dish recognition systems have been presented in (Beijbom et al., 2015), (Herranz et al., 2017), (Xu et al., 2015c). Furthermore, recent works for mobile food recognition (Oliveira et al., 2014) and mobile food calorie estimation (Okamoto and Yanai, 2016) have been proposed.

Bolaños et. al. (Bolaños et al., 2017) have proposed a deep learning system for ingredient recognition through multi-label learning. Besides, a cross-modal for recipe-retrieval have been proposed in (Chen et al., 2017a). In turn, a stacked attention network for learning the common features between the recipe image and ingredients. A joint embedding based neural network for the recipe retrieval form images and vice versa has been presented in (Salvador et al., 2017). As well as, a new large-scale dataset with 800K food images and over 1 million cooking recipes has been released in (Salvador et al., 2017). Furthermore, other food and ingredients recognition datasets are publicly available, such as, ETHZFood-101 (Bossard et al., 2014), Geolocation-food (Xu et al., 2015c), Ingredients101 and Recipes5k (Bolaños et al., 2017). However, all of these datasets are related to food and ingredients classification tasks. Since this work focuses on two main food attributes To which

country this food is related, " cuisine ", and what is the to which flavour of this food, "flavour", we have developed a new dataset for this work.

The proposed work is different from (Chen et al., 2017a), (Salvador et al., 2017) and (Bolaños et al., 2017) in such that (Chen et al., 2017a) and (Salvador et al., 2017) are mainly focused on cross-modal recipe image retrieval from food images and vice versa. In addition, the authors in (Bolaños et al., 2017) concerned on ingredients recognition through multi-label predictor for learning recipes using their own simplified dataset. As far as we know, this is the first work that attempts to classify the culinary habits from different countries with their food flavour. Thus, this work aims at developing a system for investigating cuisine and its flavour classification and for understanding food flavour. The main contributions of this work are as follows:

- To the best of our knowledge, this is the first work aims to analyze food diversity by classifying cuisine and food flavour in order to understand the food culture among the different regional peoples.
- A novel Multi-scale Convolution Network designed by aggregation of convolution layers with different kernels sizes followed by residual and pyramid pooling module with two fully connected pathway is proposed to solve the multi-modal classification problems (cuisine and flavours) with a joint weighted loss function.
- A new dataset is constructed, so-called *Yummly48K*, extracted from the *Yummly* website. Our deep model will be evaluated on the *Yummly48K* dataset.

5.2 Proposed Model

In this section, we will explain our proposed model architecture and the used joint loss function in details. The targets of our model are to predict the cuisine and its related flavour from a single input image.

5.2.1 Network Architecture

This work introduces an aggregation of convolution layers with different kernel size followed by residual and pyramid blocks with two fully connected pathway as shown in Figure 5.2.

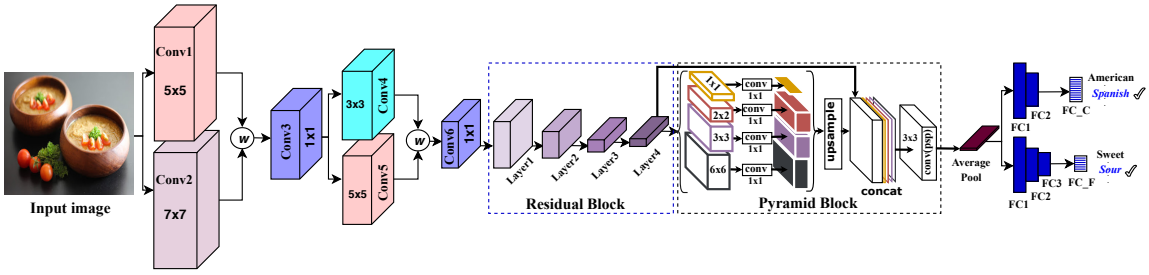


Figure 5.2: Our proposed Network Architecture.

The first layer of our proposed network is two convolutional layers with two kernel size, 5×5 and 7×7 to extract more local features of different size of neighbourhoods. To learn the best features coming from the convolutional layers, we then used an aggregation function to aggregate and weight the feature maps resulted from the first layer. A convolutional layer with kernel size, 1×1 with stride 2 is applied to reduce the size of the input image into half. Again, two convolutional layers with different kernel size, 3×3 and 5×5 are then applied. We initialized the initial convolutional layers weights are randomly. Four layers from the residual network, ResNet (He et al., 2016), are then used in the proposed network followed by a pyramid convolution layer with four levels (Zhao et al., 2017) for boosting the features into coarse-to-fine level and concatenate them together. Which enhanced the features coming from residual blocks with more details to feed the fully connected (FC layers). The weights of four layers of the residual block are used from pre-trained ResNet, and convolution layers of pyramid block are initialized randomly. Finally, two FC pathway, FC (Cuisine) and FC (Flavor) used for final classification of cuisine and flavour. FC (Cuisine) and FC (Flavor) consists of three and four FC layers with different sizes respectively. The proposed network is shown in figure 5.2 and the network architecture is detailed in Table 5.1.

Table 5.1: Architectural details of the proposed model

Blocks	Layer Name	Layer Type	K,S,P	Input Size	Output Size
Initial Blocks	Conv1	C+B+R	$5, 0, 2$	nx3x224x224	nx32x224x224
	Conv2	C+B+R	$7, 0, 3$	nx3x224x224	nx32x224x224
	W1	W*Conv1+W*Conv2	–	nx32x224x224	nx32x224x224
	Conv3	C+B+R	$1, 2, 0$	nx32x224x224	nx32x112x112
	Conv4	C+B+R	$3, 0, 1$	nx32x112x112	nx64x112x112
	Conv5	C+B+R	$5, 0, 2$	nx32x112x112	nx64x112x112
	W2	W*Conv4+W*Conv5	–	nx64x112x112	nx64x112x112
	Conv6	C+B+R	$1, 1, 0$	nx64x112x112	nx64x112x112
Residual Blocks	Layer1	Bottleneck	Bottleneck	nx64x112x112	nx256x112x112
	Layer2	Bottleneck	Bottleneck	nx256x112x112	nx512x56x56
	Layer3	Bottleneck	Bottleneck	nx512x56x56	nx1024x28x28
	Layer4	Bottleneck	Bottleneck	nx1024x28x28	nx2048x14x14
Pyramid Blocks	PSP	P+C+B+R (pool scale (1x1),(2x2),(3x3),(6x6)	$1, 0, 0$	nx2048x14x14	nx4096x14x14
	ConvPSP	C+B+R+C+B+R+D+AP	$3, 0, 1$	nx4096x14x14	n x 1024 x1x1
FC (Cuisine)	FC1	Linear 1	–	n x 1024 x1x1	n x 256 x1x1
	FC2	Linear 2	–	n x 256 x1x1	n x num_class x1x1
	FC_C	Linear 3	–	n x num_class x1x1	n x num_class
FC (Flavor)	FC1	Linear 1	–	n x 1024 x1x1	n x 512 x1x1
	FC2	Linear 2	–	n x 512 x1x1	n x 128 x1x1
	FC3	Linear 3	–	n x 128 x1x1	n x num_class x1x1
	FC_F	Linear 4	–	n x num_class x1x1	n x num_class
K= kernel size, S= stride, P= padding, C= Conv2d, B=BatchNorm2d, R=Relu, W=Weighted Aggregation Bottleneck = ResNet (He et al., 2016) Bottleneck scheme parameters, AP= average pooling, PSP= pyramid spatial pooling, FC= fully connected					

5.2.2 Multi-task Learning

Multi-modal classification problem can be solved in different ways. For example, the authors in (Bolaños et al., 2017) used binary cross-entropy loss function for multi-modal learning. They reformulated the problem as a binary classification problem. In our case, we propose to use Multi-task Learning approach to solve the multi-modal classification problem. Let L denotes the set of classes types, in this work $L = \{Cuisine, Flavor\}$. Each class type has different labels and ,thus, its own softmax classifier. We jointly train them by minimizing the multi-modal objective function defined below:

$$\ell = \sum_{i=1}^{|L|} \alpha_i \ell_i \tag{5.1}$$

where ℓ_i and α_i denote the loss function and its weight for the classification task i. The loss function ℓ_i is nothing but the categorical cross-entropy function. We observed that the numbers of instances with different labels are very unbalanced. Thus, we define ℓ_i as follows:

$$\ell_i = -\sum_{j=1}^{N} w_{y_j} y_j \log(\hat{y_j}) \tag{5.2}$$

where N is the number of instances, y_j is the actual label of the j_th instance, $\hat{y_j}$ is the prediction score, and w_{y_j}, the loss weight of the label y_j, is defined as follows:

$$w_{y_j} = 1 - \frac{N_{y_j}}{N}. \tag{5.3}$$

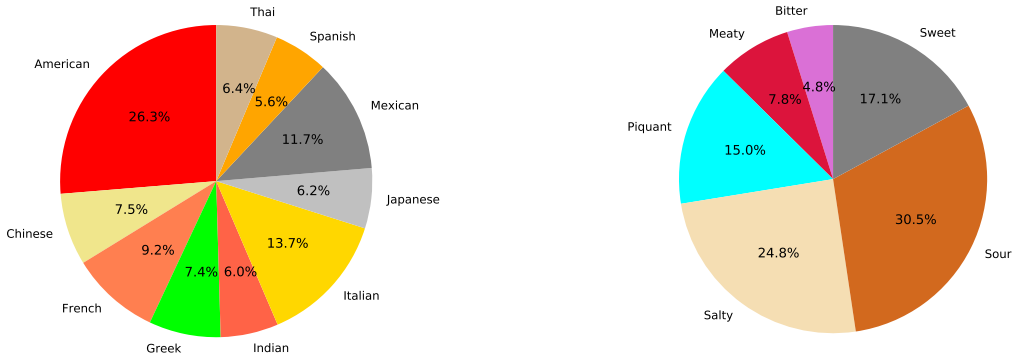
In this equation, N_{y_j} refers to the number of instances with label y_j.

5.3 Experimental Setup and Results

In this section, we describe our constructed datasets proposed for the problem of food attributes classification. In addition, we will explain the implementation of the proposed model and finally the performance of comparison between our proposed model and the baseline models, VGGNet (Simonyan and Zisserman, 2014), ResNet (He et al., 2016), and Inception-V3 (Szegedy et al., 2016).

5.3.1 Database

We constructed *Yummly48K* dataset with 48227 images that contains the information about 10 different cuisines from different countries, namely, *American, Chinese, French, Greek, Italian, Indian, Japanese, Mexican, Spanish and Thai*, in addition to 6 different flavors of the food, *Bitter, Meaty, Piquant, Salty, Sour, and Sweet*. We used python API (Gilland, 2014) for collecting our images and data from *Yummly* website. We simplified the dataset with assigning the flavours for each image taking into account only the height percentage one. For instance, an image has different flavours that are "Sweet: 0.53, Sour: 0.33, Salty: 0.16, Piquant: 0.09, Bitter: 1.0, Meaty: 0.43", we considered "Bitter" as a flavour of that image because of it provides the highest percentage of flavour in this food. The distribution of the cuisine and flavours in our dataset is presented in Figure 5.3. This dataset is divided into training (70%), validation (15%) and test (15%) sets. The original size of the collected images ranges from 200×150 to 360×240 pixels. We resized the input image by 224×224, which is the standard size of deep models for train and test.



(a) Cuisine distribution. **(b)** Flavors distribution.

Figure 5.3: Distribution of cuisine and flavors in our dataset.

5.3.2 Implementation

The proposed model is implemented on the open source deep learning library, PyTorch (Paszke et al., 2017). The Adam algorithm is used for the model optimization, which depends on the first and second-order moments of the gradient (Kingma and Ba, 2014). In addition, a "poly" learning rate policy is used for adjusting learning rate and selected a base learning rate of 0.001 with a power of 0.9 (Chen et al., 2016a). For data augmentation, we selected random cropping, random horizontal and vertical rotation between -10 and 10 degrees. The "batch size" is set to 16 for training and the number of epochs to 100. The experiments utilized NVIDIA TITAN X with 12GB memory and it takes approximately 3 days to train the networks.

5.3.3 Results and discussion

To evaluate our model, we used standard evaluation metrics; *Precision, Recall and* F_1 *score* that are commonly used in the image classification task. We compare the proposed model with common baseline models. The baseline tested models have been updated for the multi-modal (MM) classification task to work on our dataset, *Yummly48K*, by using two fully-connected (FC) layers for two our targets, cuisine and flavour, instead of one FC layer used at the classical classification

Table 5.2: Multi-Modal classification results on our dataset

Models	Validation			Test		
	Precision	Recall	F_1 score	Precision	Recall	F_1 score
VGG-16 (MM)	38.12	25.06	30.24	36.46	24.85	29.55
ResNet-50 (MM)	61.30	49.04	54.48	59.07	47.44	52.62
Inception-V3 (MM)	63.91	52.13	57.42	61.39	50.51	55.42
Proposed	**72.33**	**59.53**	**65.37**	**69.54**	**57.19**	**62.76**

models, VGGNet (Simonyan and Zisserman, 2014), ResNet (He et al., 2016), and Inception-V3 (Szegedy et al., 2016). The performance of the comparison is shown in Table 7.4. All measures reported in % and the best results are highlighted in boldface. We calculate the average of cuisine and flavours metrics on our validation and test dataset.

Figure 5.4: Some examples of correctly classify both cuisine and flavor label (all image on upper row), correctly predicted cuisine, but incorrectly predicted flavor label (lower row 1^{st} and 2^{nd} image), incorrectly classify both cuisine and flavor label (lower row 3^{rd} and 4^{th} image) (**GD:** ground truth, **PD:** predictions).

Some examples of our experimental results are shown in Figure 5.4. We observed that the misclassification is occurred by our model in Italian and Spanish cuisine, whose main ingredient is pasta. Similarly, between Thai and Chinese has some common features, so it also can misclassify some cuisine from this region, although our model can correctly identify the flavour of it. However, the model can not distinguishes between Mexican *"Burritos"* with Greek *"Burritos"* and also misclassify the flavour of *"Burritos"*. Likewise, some French cuisine misclassified to Spanish and also the flavor.

5.4 Conclusion

The food culture has a strong effect on everyday life and it reflects the person's history, lifestyle, values, and beliefs from different countries. In this chapter, we presented cuisine and flavours classification methods by the multi-scale convolutional network to identify from a food image. A feature maps aggregation is also used for improving the network performance. Besides, this work provided a new dataset for food attributes classification. The proposed model achieved an acceptable classification rate comparing with recent state-of-the-art models. The direction of our future research hints to continue with the fusion of the Recurrent Neural Networks. Furthermore, we aim at increasing food attributes to classify cuisine, course, nutrition's, ingredients and flavours to develop a unified AI framework of food attributes analysis.

Chapter 6

SLSDeep: Skin Lesion Segmentation Based on Dilated Residual and Pyramid Pooling Networks

6.1 Introduction

According to the Skin Cancer Foundation statistics, the percentage of both melanoma and non-melanoma skin cancers is rapidly being increased over the last few years (Siegel et al., 2017). Dermoscopy, non-invasive dermatology imaging methods, can help the dermatologists to inspect the pigmented skin lesions and diagnose malignant melanoma at an initial-stage Kardynal and Olszewska (2014). Even the professional dermatologists can not properly classify the melanoma only by relying on their perception and vision. Sometimes human tiredness and other distractions during visual diagnosis can also yield a high number of false positives. Therefore, a Computer-Aided Diagnosis (CAD) system is needed to assist the dermatologists to properly analyze the dermoscopic images and accurately segment the melanomas. Many melanoma segmentation approaches have been proposed in the literature. An overview on numerous melanoma segmentation techniques is presented in (Zhang, 2017). However, this task is still a challenge, since the dermoscopic images have various complexities including different sizes and shapes, fuzzy boundaries, different

colors and the presence of hair Day and Barbour (2000).

In last few decades, many approaches have been proposed to cope with the aforementioned challenges. Most of them are based on thresholding, edge-based, region-based active contour models, clustering and supervised learning Celebi et al. (2015). However, these methods are unreliable when dermoscopic images are inhomogeneous or lesions have fuzzy or blurred boundaries (Celebi et al., 2015). Furthermore, their performance relies on efficient pre-processing algorithms, such as illumination correction and hair removal, which badly affect the generalizability of these models.

Recently, deep learning methods applied to image analysis, specially Convolutional Neural Networks (CNNs) have been used to solve the image segmentation problemLong et al. (2015). These CNN-based methods can automatically learn features from raw pixels to distinguish between background and foreground objects to attain the final segmentation. Most of these approaches generally are based on encoder-decoder networks (Long et al., 2015). The encoder networks are used for extracting the features from the input images, in turn the decoder ones used to construct the segmented image. The U-net network proposed in (Ronneberger et al., 2015) has been particularly designed for biomedical image segmentation based on the concept of Fully Convolutional Networks (FCN) (Long et al., 2015). The U-net model reuses the feature maps of the encoder layers to the corresponding decoders and concatenates them to upsampled decoder feature maps, which are also called "skip-connections". The U-Net model for SLS outperformed many classical clustering techniques (Lin et al., 2017a).

In addition, the deep residual network (ResNet) model (Yu et al., 2017b) is a 50-layers network designed for segmentation tasks. ResNet blocks are used to boost the overall depth of the networks and allow more accurate segmentation depending on more significant image features. Moreover, Dilated Residual Networks (DRNs) proposed in (Yu et al., 2017a) increase the resolution of the ResNet blocks's output by replacing a subset of interior subsampling layers by dilation (Yu and Koltun, 2015). DRNs outperform the normal ResNet without adding algorithmic complexity to the

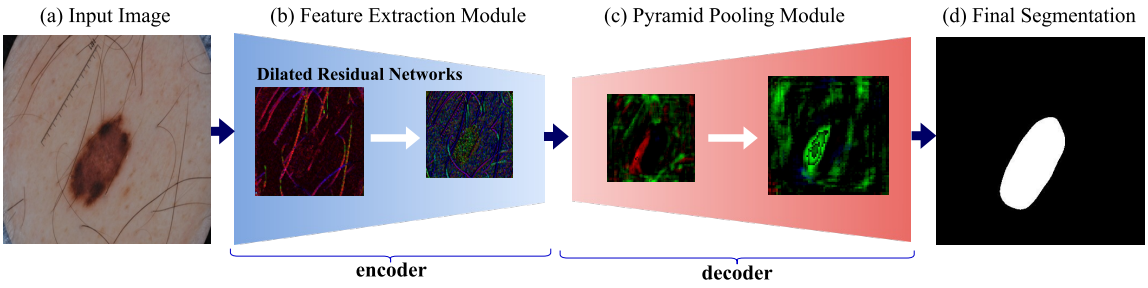


Figure 6.1: Architecture of the proposed skin lesion segmentation network.

model. DRNs are able to represent both tiny and large image features. Furthermore, a Pyramid Pooling Network (PPN) that is able to extract additional contextual information based on a multi-scale scheme is proposed for image segmentation (Zhao et al., 2017).

Inspired by the success of the aforementioned deep models for semantic segmentation, we propose a model combining skip-connections, dilated residual and pyramid pooling networks for SLS with different improvements. In our model, the encoder network depends on DRNs layers, in turn the decoder depends on a PPN layer along with their corresponding connecting layers. More features can be extracted from the input dermoscopic images by combining DRNs with PPN, in turn it also enhances the performance of the final network. Finally, our SLS segmentation model uses a new loss function, which combines Negative Log Likelihood (NLL) and End Point Error (EPE) (Baker et al., 2011). Mainly, cross-entropy is used for multi-class segmentation models, however it is not as useful as NLL in binary class segmentation. Thus, in such melanoma segmentation, we propose to use NLL as a loss function. In addition, for preserving the melanoma boundaries, EPE is used as a content loss function. Consequently, this work aims at developing an automated deep SLS model with two main contributions:

- An encoder-decoder network for efficient SLS without any pre- and post-processing algorithms based on dilated residual and pyramid pooling networks to enclose coarse-to-fine features of dermoscopic images.
- A new loss function that is a combination of Negative Log Likelihood and End Point Error for properly detecting the melanoma with weak edges.

6.2 Proposed Model

6.2.1 Network Architecture

Figure 6.1 shows the main framework of the proposed SLSDeep model and Figure 6.2 illustrates the architecture of the model with DRNs (Zhou et al., 2017) and PPN (He et al., 2015b). The network contains two-fold architecture: encoder and decoder. Regarding the encoder phase, the first layer is a 3×3 convolutional layer followed by 3×3 max pooling with stride 2.0 that generates 64 feature maps. This layer uses ReLU as an activation and batch normalization to speed-up the training steps with a random initialization. Following, four pre-trained DRNs blocks are then used to extract 256, 512, 1024 and 2048 feature maps, respectively as shown in Figure 6.2. The first, third, and fourth DRNs layers are with stride 1.0, in turn the second one is with stride 2.0. Thus, the size of final output of encoder is 1/8 of the input image (e.g. in our model, the input image is in 384×384 and the output feature maps of the encoder is 48×48). For global contextual prior, average pooling is used before feeding to fully connected layers in image classification Szegedy et al. (2015). However, it is not sufficient to extract necessary information from our skin lesion images. Therefore, we do not use average pooling at the end of the encoder and directly fed the output feature maps to the decoder network. On the other hand, for the decoder network, we use the concept of PPN for producing multi-scale (coarse-to-fine) feature maps and then all scales are concatenated together to get more robust feature maps. PPN use a hierarchical global prior of variant size feature maps in multi-scales with different spatial filters as shown in Figure 6.2. In this work,

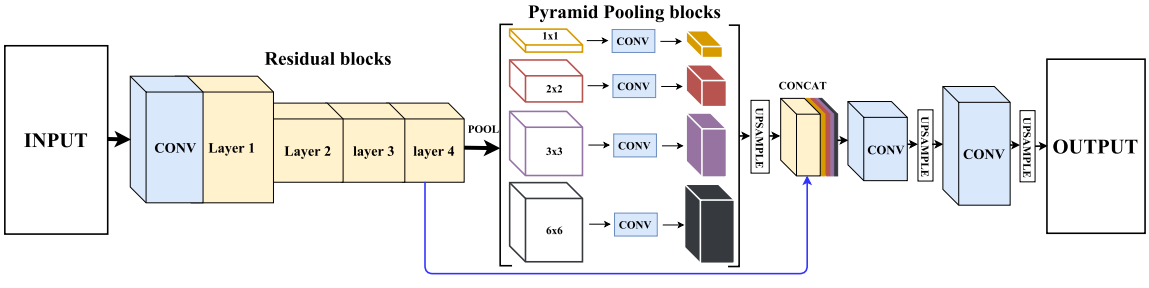


Figure 6.2: Architecture of the encoder-decoder network.

the used PPN layer extracts feature maps using four pyramid scales with rescaling sizes of $1 \times 1, 2 \times 2, 3 \times 3$ and 6×6. A convolutional layer with a 1×1 kernel in every pyramid level is used for generating 1024 feature maps. The low-dimension feature maps are then upsampled based on bilinear interpolation to get the same size of the input feature maps. The input and four feature maps are finally concatenated to produce 6144 feature maps (i.e., 4x1024 feature maps concatenated with the input 2048 feature maps). Sequentially, two 3×3 convolutional layers are followed by two upsampling layers. Finally, a softmax function (i.e. normalized exponential function) is utilized as logistic function for producing the final segmentation map. A ReLU activation with batch normalization is used in the two convolutional layers (Ioffe and Szegedy, 2015). Moreover, in order to avoid the overfitting problem, the dropout function with a ratio of 0.5 (Srivastava et al., 2014) is used before the second upsampling layer. The skip connections between all layers of the encoder and decoder were tested during the experiments. However, the best results were provided when only one connection was skipped between the last layer of the encoder and the output of PPN layer of the decoder. The details of the encoder and decoder architectures are given in the supplementary materials.

6.2.2 Loss Function

Most of the traditional deep learning methods commonly employ cross-entropy as a loss function for segmentation (Ronneberger et al., 2015). Since the melanoma is mostly a small part of a dermoscopic image, the minimization of cross-entropy tends to be biased towards the background. To cope with this challenge, we propose a new loss function by combining objective and content losses: NLL and EPE, respectively. In order to fit a log linear probability model to a set of binary labeled classes, the NLL that is our objective loss function is minimized.

Let $v \in \{0, 1\}$ be a true label for binary classification and $p = Pr(v = 1)$ a probability estimate, the NLL of the binary classifier can be defined as:

$$L_{nll}(v, p) = -\log Pr(v|p) = -(v \log(p) + (1 - v) \log(1 - p)). \tag{6.1}$$

In order to maximize Peak Signal-to-Noise Ratio, a content loss function based on an end-point error proposed in (Baker et al., 2011) is used for preserving the melanoma boundaries. In EPE, We compared the magnitude and orientation of the edges of the predicted mask with the correct one. Let M a generated mask and G the corresponding ground-truth, then the EPE can be defined as:

$$L_{epe} = \sqrt{(M_x - G_x)^2 + (M_y - G_y)^2}, \tag{6.2}$$

where (M_x, M_y) and (G_x, G_y) are the first derivatives of M and G, respectively in x and y directions.

Thus, our final loss function combining the NLL and EPE can be defined as:

$$L_{total} = L_{nll} + \alpha L_{epe}, \tag{6.3}$$

where $\alpha < 1$ is a weighted coefficient. In this work, we use $\alpha = 0.5$.

6.3 Experimental Setup and Evaluation

Database: To test the robustness of the proposed model, it was evaluated on two public benchmark datasets of dermoscopy images for skin lesion analysis: **ISBI 2016** (Codella et al., 2017) and **ISBI 2017** (Gutman et al., 2016). The datasets images are captured by different devices at various top clinical centers around the world. In ISBI 2016 dataset, training and testing part contain 900 and 379 annotated images, respectively. The size of the images ranges from 542×718 to 2848×4288 pixels. In turn, ISBI 2017 dataset is divided into training, validation and testing parts with 2000, 150 and 600 images, respectively.

Evaluation Metrics: We used the evaluation metrics of ISBI 2016 and 2017 challenges for evaluating the segmentation performances including Specificity(SPE), Sensitivity(SEN), Jaccard index(JAC), Dice coefficient(DIC) and Accuracy(ACC) detailed in Gutman et al. (2016) and Codella et al. (2017).

Implementation: The proposed model is implemented on an open source deep

learning library named PyTorchPaszke et al. (2017). For optimization algorithm, we used Adam Kingma and Ba (2014) for adjusting learning rate, which depends on first and second order moments of the gradient. We used a "poly" learning rate policy Chen et al. (2016a) and selected a base learning rate of 0.001 and 0.01 for encoder and decoder, respectively with a power of 0.9. For data augmentation, we selected random scale between 0.5 and 1.5, random rotation between -10 and 10 degrees. The "batchsize" is set to 16 for training and the epochs to 100. All the experiments are executed on NVIDIA TITAN X with 12GB memory taking around 20 hours to train the network.

Evaluation and results: Since the size of the given images is very large, we resized the input images to 384×384 pixels for training our model. In this work, we tested different sizes and the 384×384 size yields the best results. In order to separately assess the different contributions of this model, the resulting segmentation for the proposed model with different variations have been computed: (a) The SLSDeep model without the content loss EPE (SLSDeep-EPE), (b) the proposed method with skip connections of all encoder and decoder layers (SLSDeep+ASC) and (c) the final proposed model (SLSDeep) with NLL and EPE loss functions and only one skip connection between the last layer of the encoder and the PPN layer. Quantitative results on ISBI'2016 and ISBI'2017 datasets are shown in Table 6.1. Regarding ISBI'2016, we compared the SLSDeep and its variations to the four top methods: ExB, Yu et al. (2017b), Rahman et al. (2016) and Yuan (2017) providing the best results according to Gutman et al. (2016). The segmentation results of our model SLSDeep with its variations (SLSDeep-EPE and SLSDeep+ASC) provided better results than the other four evaluated methods on the ISBI'2016 in terms of the five aforementioned evaluation metrics. SLSDeep yields the best results among the three variations. In addition, for the DIC score, our model, SLSDeep, improved the results with around 3.5%, while the JAC score was significantly improved with 8%. The SLSDeep yielded results with an overall accuracy of more than 98%. Furthermore, SLSDeep on the ISBI'2017 provided segmentation results with improvements of 3% and 2% in terms of DIC and JAC scores, respectively. Again

Table 6.1: Performance Evaluation on the ISBI Challenges Dataset

Challenges	Methods	ACC	DIC	JAC	SEN	SPE
ISBI 2016	ExB	0.953	0.910	0.843	0.910	0.965
	CUMEDYu et al. (2017b)	0.949	0.897	0.829	0.911	0.957
	Rahman et. al.Rahman et al. (2016)	0.952	0.895	0.822	0.880	0.969
	Yuan et. al.Yuan (2017)	0.955	0.912	0.847	0.918	0.966
	SLSDeep	**0.984**	**0.955**	**0.913**	**0.945**	**0.992**
	SLSDeep-EPE	0.973	0.919	0.850	0.890	0.990
	SLSDeep+ASC	0.975	0.930	0.869	0.952	0.979
ISBI 2017	Yuan et. al.Yuan (2017)	0.934	0.849	0.765	**0.825**	0.975
	Berseth et. al.Berseth (2017)	0.932	0.847	0.762	0.820	0.978
	MResNet-SegBi et al. (2017)	0.934	0.844	0.760	0.802	0.985
	SLSDeep	**0.936**	**0.878**	**0.782**	0.816	0.983
	SLSDeep-EPE	0.913	0.826	0.704	0.729	**0.986**
	SLSDeep+ASC	0.906	0.850	0.739	0.808	0.905

SLSDeep outperformed the three top methods of the ISBI'2017 benchmark, (Yuan, 2017), Berseth (2017) and Bi et al. (2017), in terms of ACC, DIC and JAC scores. However, Yuan (2017) yielded the best SEN score with just a 0.9% improvement than our model. The SLSDeep-EPE and SLSDeep+ASC provided reasonable results, however their results were worse than the other tested methods in terms of ACC, DIC, JAC and SEN. However, SLSDeep-EPE yields the highest SPE with a 0.1% and 0.3% more than MResNet-Seg Bi et al. (2017) and SLSDeep, respectively. Using the EPE function with the final SLSDeep model significantly improved the DIC and JAC scores of 3% and 5%, respectively, on ISBI'2016 and of 5% and 8%, respectively, with ISBI'2017. In addition, SLSDeep with only one skip connections yields better results than SLSDeep+ASC on both ISBI datasets.

Qualitative results of four examples from the ISBI'2017 dataset are shown in Figure 6.3. For the first and second examples (on the top- and down-left side), the lesions were properly detected, although the color of the lesion area is very similar to the rest of the skin. In addition, the lesion area was accurately segmented regardless the unclear melanoma edges. Regarding the third example (on the top-right side), SLSDeep properly segmented the lesion area; however a small false region having similar melanoma features was also detected. The last example is very tricky, since the lesion shown is very small. However, the SLSDeep model was able to detect it, but with a large false negative region.

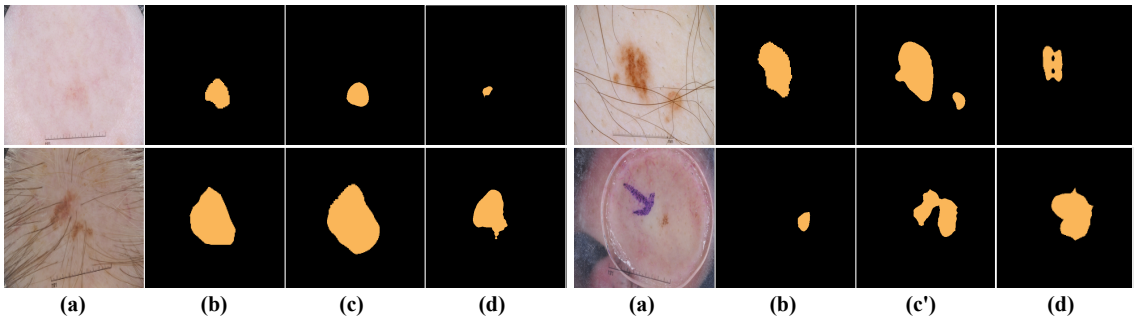


Figure 6.3: Segmentation results: (a) input image, (b) ground truth and (c) correct segmentation by our model, (c') incorrect segmentation by our model, (d) segmentation by Yuan et al. Yuan (2017) model.

6.4 Conclusions

This chapter proposed a novel deep learning skin lesion segmentation model based on training an encoder-decoder network. The encoder network used the dilated ResNet layers with downsampling to extract the features of the input image, in turn convolutional layers with pyramid pooling and upsampling are used to reconstruct the segmented image. This approach outperforms, in terms of skin lesion segmentation, the literature evaluated on two ISBI'2016 and ISBI'2017 datasets. The quantitative results show that SLSDeep is a robust segmentation technique based on different evaluation metrics: accuracy, Dice coefficient, Jaccard index and specificity. In addition, qualitative results show promising skin lesion segmentation. Future work aims at applying the proposed model to various medical applications to prove its versatility.

Chapter 7

MobileGAN: Skin Lesion Segmentation Using a Lightweight Generative Adversarial Network

7.1 Introduction

Skin cancer is one of the wide speared of cancers. According to World Health Organization (WHO),there are 1.04 million cases in 2018 (WHO, 2018a). Over the last decades, the percentage of both melanoma and non-melanoma skin cancers increased rapidly (Apalla et al., 2017). Melanoma is the most dangerous types of skin cancer, and 75% of deaths are related to it (Atlanta, 2011). Image analysis techniques (Dermoscopy) based on computerized non-invasive dermatology is getting very important for physicians to inspect the pigmented skin lesions and detect malignant melanoma at an early stage (Esteva et al., 2017) in order to improve the survival rate and reduce cost. Consequently, a Computer-Aided Diagnosis (CAD) system is essential to support the dermatologists to investigate the dermoscopic images and segment melanomas as precisely as possible. Several melanoma segmentation methods have been proposed in the literature (Al-Masni et al., 2018). The main challenges faced in the segmentation of pigmented skin lesion include the huge diversity in color, shape, texture, size, but also the low contrast between skin

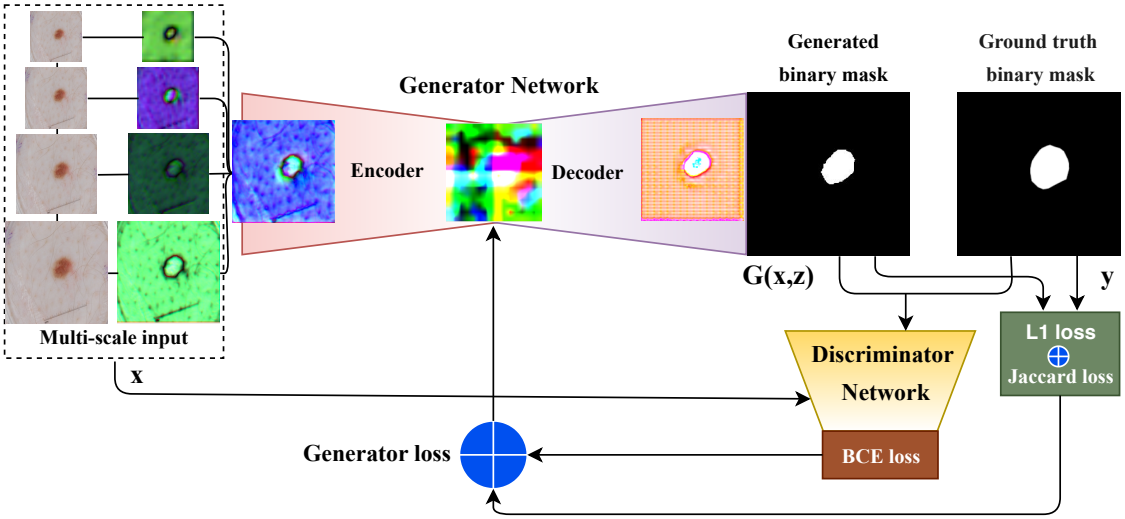


Figure 7.1: The framework of the proposed MobileGAN

tissues, the irregular and fuzzy boundaries and the presence of blood vessels and hairs (Al-Masni et al., 2018). Several methods have been proposed to cope with these challenges using traditional image processing algorithms, such as histogram thresholding, unsupervised clustering, and supervised segmentation methods (see an overview in (Celebi et al., 2015)). However, these approaches yield inaccurate segmentation results, when the skin lesions have fuzzy boundaries (Celebi et al., 2015). In addition, the performance of these methods highly relies on pre-processing algorithms, such as hair removal and contrast enhancement. With the rapid progress in deep learning models, many skin lesion segmentation approaches have been introduced increasing the accuracy of segmentation. For instance, the SLSDeep model was proposed in (Sarker et al., 2018b) to segment the skin lesion by using feature pyramid pooling. In (Al-Masni et al., 2018), a full resolution convolutional networks (FrCN) was introduced to directly learn the full resolution features of each pixel of the input image without the need for pre- or post-processing operations. Besides, GAN with a multi-scale loss function, called SegAN, has also been proposed for skin lesion segmentation in (Xue et al., 2018).

All of the methods mentioned above provided high precision. However, they have tens or hundreds of millions of parameters. In this work, we propose a lightweight GAN model, named MobileGAN, for skin lesion segmentation

of dermoscopic images. In the proposed model, we extract low features with multi-scale convolutional networks. In order to reduce the computational cost, the proposed model uses 1D non-bottleneck factorization network. Moreover, position and channel attention modules are used to improve the features representation regardless of spatial and channel dimensions.

The main contribution are:

- To cope with shadows by supposing that only the part of true lesions appears at multiple scales (consequently, a multi-scale block is introduced for aggregating the coarse-to-fine features of dermoscopic images).
- To reduce the computational cost by using a 1D non-bottleneck factorized network (Romera et al., 2018).
- To enhance the discriminant ability of feature representations in spatial and channel dimensions by using both position and channel attention models (Fu et al., 2018).
- To use a combination of the binary cross entropy, Jaccard and L_1-norm as a loss function for training the modified GAN model.

7.2 Proposed Model

7.2.1 Network architecture

The generative adversarial network pix2pix (Isola et al., 2017) has been used in different tasks, such as synthetic image generation and medical image segmentation. It consists of two main networks: generator G and discriminator D. The generator is an encoder-decoder architecture that learns the mapping from an image from domain A (the skin image) to domain B (the segmented lesions). The discriminator compares the generated segmentation masks with real segmented images. Figure 7.2 presents the architecture of the proposed model, which has the G and D networks as the pix2pix model. We remind that, to alleviate false detection due to shadows, a multi-scale block for aggregating the coarse-to-fine features of dermoscopic images

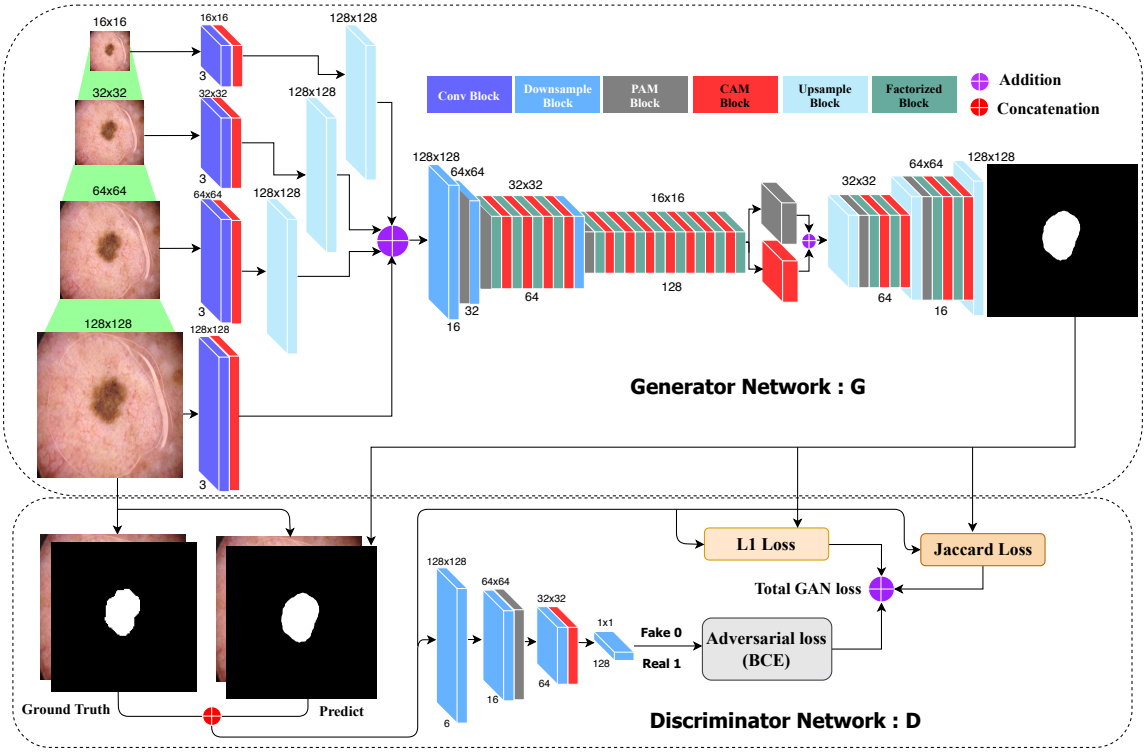


Figure 7.2: The architecture of the proposed MobileGAN network: generator network (**top**) and discriminator network (**bottom**).

is used. Below, we explain the encoder and decoder networks of the generator, and the discriminator networks in details.

7.2.1.1 The Encoder Network

the input images for the encoder of generator network G are scaled to four resolutions (i.e., the original input size and three different resolutions) as shown in Figure 7.2. The four resolutions are feed into four convolution blocks to generate 4×16 feature maps. The four convolutional blocks are then followed by four channel attention module (CAM) to capture visual features dependencies in channel dimensions(for more details, see the section CAM).Afterward, we upsample three scaled inputs to the same size of the original input image by using bilinear interpolation and then average all feature maps of the four scales to generate 1×16 feature maps. The encoder network can extract low features in different scales in order to cope with shadows. In addition, the resulted feature maps are created in both spatial and frequency domains. The resulted 16 feature maps are fed into two

Convolutional-Downsampling-Attention (CDA) layers. Each CDA layer comprises a convolutional block followed by a max pooling of 2, and then a Position Attention Module (PAM) to capture the spatial features(for more details, see the section PAM). The two layers produce 64 feature maps that are fed into the next four factorized-attention (FCA) layers (for more details, see the section FCA layer). Each FCA layer consists of a non-bottleneck factorized block followed by a CAM. The resulting feature maps are fed into a CDA layer to obtain 128 feature maps that are fed into eight FCA layers. The result of the eighth FCA layer is fed to a non-bottleneck factorized block followed by two parallel attention blocks; one for CAM and the other for PAM that is summed to capture visual high features independently to position and channel dimensions. The final 128 feature maps are fed into the decoder to construct the segmented image.

Channel Attention Module(CAM): Every channel map of high-level features can be noted as a class-specific response, and various semantic responses are correlated with each other. Using the interdependencies among channel maps, we could highlight interdependent feature maps and update the feature representation of specific interpretation. Therefore, we build a channel attention module to explicitly model interdependencies among channels. The composition of channel attention module is shown in Figure. 7.3(B). The channel attention map $\mathbf{X} \in \mathbb{R}^{C\times C}$ from the original features $\mathbf{A} \in \mathbb{R}^{C\times H\times W}$ directly calculate from the position attention module. Clearly, it reshape $\mathbf{A}$ to $\mathbb{R}^{C\times N}$ then perform a matrix multiplication between $\mathbf{A}$ and the transpose of $\mathbf{A}$. Eventually, a softmax function is applied to generate the channel attention map $\mathbf{X} \in \mathbb{R}^{C\times C}$:

$$x_{ji} = \frac{exp(A_i \cdot A_j)}{\sum_{i=1}^{C} exp(A_i \cdot A_j)} \tag{7.1}$$

where x_{ji} calculates the i^{th} channel's impact on the j^{th} channel. Moreover, a matrix multiplication between the transpose of $\mathbf{X}$ and $\mathbf{A}$ and reshape their result to $\mathbb{R}^{C\times H\times W}$ is used. Afterward, the result multiply by a scale parameter β and perform an element-wise addition operation with $\mathbf{A}$ to get the final output $\mathbf{E} \in \mathbb{R}^{C\times H\times W}$:

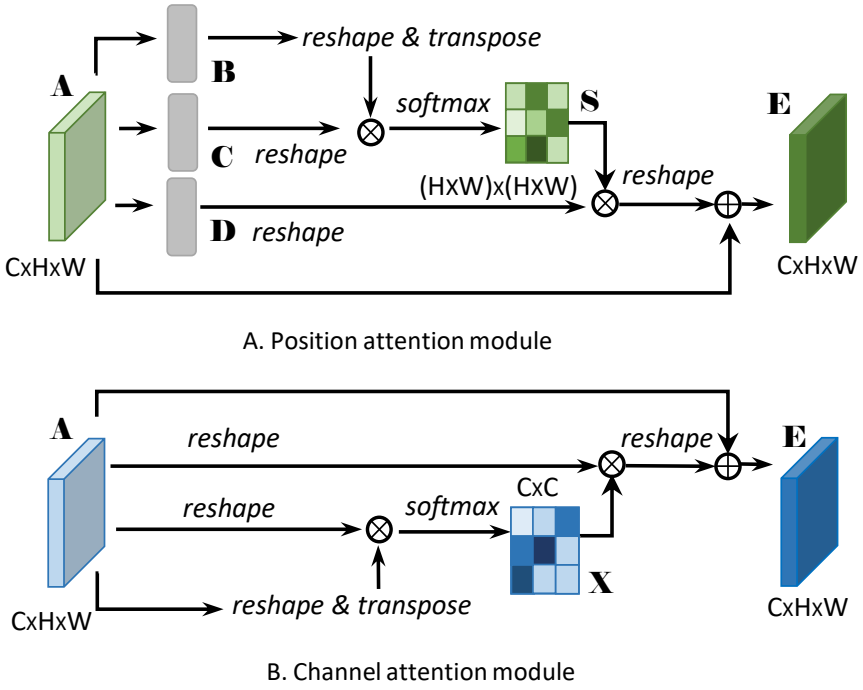


Figure 7.3: Architecture of PAM and CAM module (Fu et al., 2018).

$$E_j = \beta \sum_{i=1}^{C} (x_{ji} A_i) + A_j \tag{7.2}$$

where β continuously learn a weight from 0. The final feature of each channel is a weighted sum of the features of all channels and original features, which models the long-range semantic dependencies between feature maps shown in the Equation 7.2. It highlights class-dependent feature maps and supports to boost feature discriminability. Regarded, the convolution layers to set features before estimating correlations of two channels, considering it can keep the association between different channel maps are not employed. Thus, recent work (Zhang et al., 2018) in which investigates channel relationships by a global pooling or encoding layer. Here, spatial information at all corresponding positions to model channel correlations is used.

Position Attention Module (PAM): Discriminant feature descriptions are necessary for skin lesion segmentation, which could be achieved by capturing long-range contextual information. In order to model strong contextual links over local feature descriptions, a position attention module used. The position attention

module encodes a comprehensive series of contextual information into local features, therefore improving their representational role. Following, the method to adaptive aggregate spatial contexts are refined.

Given a local feature $\mathbf{A} \in \mathbb{R}^{C\times H\times W}$, feed into a convolution layers with batch normalization and ReLU to produce two new feature maps $\mathbf{B}$ and $\mathbf{C}$, respectively, where $\{\mathbf{B}, \mathbf{C}\} \in \mathbb{R}^{C\times H\times W}$ shown in Figure.7.3(A). Afterwards, resize them to $\mathbb{R}^{C\times N}$, where $N = H \times W$ is the number of features. Later, performed a matrix multiplication between the transpose of $\mathbf{C}$ and $\mathbf{B}$, and use a softmax function to estimate the spatial attention map $\mathbf{S} \in \mathbb{R}^{N\times N}$:

$$s_{ji} = \frac{exp(B_i \cdot C_j)}{\sum_{i=1}^{N} exp(B_i \cdot C_j)} \tag{7.3}$$

where s_{ji} means the i^{th} position's contact on j^{th} position. Remark that the further related feature descriptions of the two position give to a higher correlation between them. However, the feature $\mathbf{A}$ feed into a convolution layer with batch normalization and ReLU to create a new feature map $\mathbf{D} \in \mathbb{R}^{C\times H\times W}$ and resize it to $\mathbb{R}^{C\times N}$. Next, a matrix multiplication between $\mathbf{D}$ and the transpose of $\mathbf{S}$ and reshape the result to $\mathbb{R}^{C\times H\times W}$ has been performed. Lastly, a scale parameter α multiply with it and perform an element-wise addition operation with the features $\mathbf{A}$ to get the final output $\mathbf{E} \in \mathbb{R}^{C\times H\times W}$ as follows:

$$E_j = \alpha \sum_{i=1}^{N} (s_{ji} D_i) + A_j \tag{7.4}$$

where α is initialized as 0 and continuously study to assign more weight (**?**). It can be concluded from Equation 7.4 that the resulting feature $\mathbf{E}$ at every position is a weighted sum of the features at complete positions and original features. Consequently, it has a global contextual representation and selectively aggregates contexts according to the spatial attention map. The related semantic features achieve mutual gains, thus improving intra-class compact and semantic consistency.

Factorized-attention (FCA) Block: In MobileGAN, we use residual 1-D kernel factorization attention module is used for reducing the computation

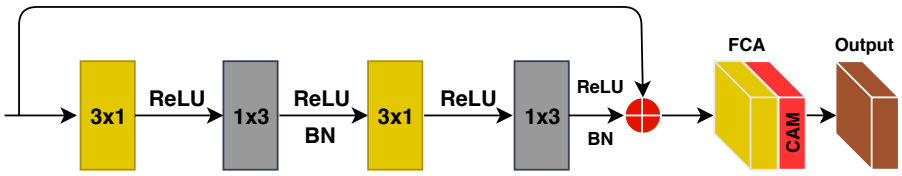


Figure 7.4: Architecture of Factorized-attention (FCA) module.

complexity. Assume, $\mathbf{W} \in \mathbb{R}^{C \times d^h \times d^v \times F}$ denote the weights of a typical 2D convolutional layer, where C is the number of input planes, F is the number of output planes (feature maps) and $d^h \times d^v$ indicates the kernel size of every feature map (typically $d^h \equiv d^v \equiv d$). Let $b \in \mathbb{R}^F$ be the vector denoting the bias term for every filter and $\mathbf{f^i} \in \mathbb{R}^{d^h \times d^v}$ denote the i^{th} kernel in the layer. It is possible to relax the rank-1 constraint and essentially rewrite $\mathbf{f^i}$ as a linear combination of 1D filters:

$$\mathbf{f^i} = \sum_{k=1}^{K} \sigma_k^i \bar{v}_k^i \left(\bar{h}_k^i\right)^T \tag{7.5}$$

where $\bar{v}_k^i$ and $\left(\bar{h}_k^i\right)^T$ are vectors of length d, σ_k^i is a scalar weight and K is a rank of $\mathbf{f^i}$. Thus, the ith output of the decomposed layer, a_i^1 can be expressed as a function of its input a_*^0, in the following way:

$$a_i^1 = \varphi\left(b_i^h + \sum_{l=1}^{L} \bar{h}_{il}^T * \left[\varphi\left(b_l^v + \sum_{c=1}^{C} \bar{v}_{lc} * a_c^0\right)\right]\right) \tag{7.6}$$

where $\varphi(.)$ represent the non-linearity of the 1D decomposed filters, which can be implemented with ReLU. Figure 7.4 shows the factorized block diagram. Finally, we fed it to the channel attention module and get the final representation of the factorized-attention (FCA) module.

7.2.1.2 The Decoder Network:

We upsample the final output of the encoder to feed both streams. Each stream consists of one Deconvolutional-Upsampling-Attention (DUA) and two FCA layers. The final feature maps are upsampled to obtain the segmented image. In all layers of the encoder and decoder networks, we used convolutional and deconvolutional filters with a kernel size of 3×3, a stride of 2 and a padding of 1 (for more details, see the

Table 7.1). In the testing phase, the trained generator network G is used to produce the segmentation mask for each test image.

7.2.1.3 The Discriminator Network:

It comprises four convolutional and downsampling layers. The four convolutional layers use a kernel of 4×4, a stride of 2, and a padding of 1. In the second layer, a PAM block is added after the convolutional block, while in the third layer, a CAM block is added.

Table 7.1: Proposed MobileGAN Network Architecture.

		Layer number	Block	Type	Parameters					Input Size	Output Size
					K	S	P	Di	Dr		
Gneerator	Encoder	1	Conv2D	C	16	1	1	0	0	nx3x128x128	nx16x128x128
				C+UP (1)							
				C+UP (2)							
				C+UP (3)							
		2	CDA	C	3	2	1	0	0	nx16x128x128	nx32x64x64
				M+BN	2	2	0	0	0		
		3	CDA	C	3	2	1	0	0	nx32x64x64	nx64x32x32
				M +BN	2	2	0	0	0		
		4-8	5 x FCA	1D-F	(3,1), (1,3)	1	(1,0), (0,1)	(1,1), (1,1)	0.3	nx64x32x32	nx64x32x32
		9	CDA	C	3	2	1	0	0	nx64x32x32	nx128x16x16
				M+BN	2	2	0	0	0		
		10	FCA	1D-F	(3,1), (1,3)	1	(2,0), (0,2)	(2,1), (1,2)	0.3	nx128x16x16	nx128x16x16
		11	FCA	1D-F	(3,1), (1,3)	1	(4,0), (0,4)	(4,1), (1,4)	0.3		
		12	FCA	1D-F	(3,1), (1,3)	1	(8,0), (0,8)	(8,1), (1,8)	0.3		
		13	FCA	1D-F	(3,1), (1,3)	1	(16,0), (016)	(16,1), (1,16)	0.3		
		14	FCA	1D-F	(3,1), (1,3)	1	(2,0), (0,2)	(2,1), (1,2)	0.3		
		15	FCA	1D-F	(3,1), (1,3)	1	(4,0), (0,4)	(4,1), (1,4)	0.3		
		16	FCA	1D-F	(3,1), (1,3)	1	(8,0), (0,8)	(8,1), (1,8)	0.3		
		17	FCA	1D-F	(3,1), (1,3)	1	(16,0), (016)	(16,1) (1,16)	0.3		
		18	CAM+PAM	CAM Module + PAM Module						nx128x16x16	nx128x16x16
	Decoder	19	DUA	CT+ BN	3	2	1	0	0	nx128x16x16	nx64x32x32
		20-21	FCA	1D-F	(3,1), (1,3)	1	(1,0), (0,1)	(1,1), (1,1)	0	nx64x32x32	nx64x32x32
		22	DUA	CT+ BN	3	2	1	0	0	nx64x32x32	nx16x64x64
		23-24	FCA	1D-F	(3,1), (1,3)	1	(1,0), (0,1)	(1,1), (1,1)	0	nx16x64x64	nx16x64x64
		25	UP	CT+ Tanh	3	2	0	0	0	nx16x64x64	nx3x128x128
Discriminator		1	CD	C+ BN+LR (0.2)	4	2	1	0	0	nx6x128x128	nx16x64x64
		2	DN+PAM	C+ BN+LR (0.2)	4	2	1	0	0	nx16x64x64	nx64x32x32
		3	DN+CAM	C+ BN+LR (0.2)	4	1	1	0	0	nx64x32x32	nx128x31x31
		4	DN	C+Sigmoid	4	1	0	0	0	nx128x31x31	nx1x30x30

CDA = Convolutional-downsampling-attention, FCA = Factorized-attention, DUA = Deconvolutional-upsampling-attention, 1D-F = 1D non-bottleneck factorization, CD = Convolutional-downsampling, UP = Upsample, DN = Downsample, BN = BatchNorm2d,C = Conv2D, CT = ConvTranspose2d, M = Maxpool, LR = LeakyReLU, K= Kernel size, S = Stride, P = Padding, Di = Dilation, Dr = Dropout, CAM = Channel-attention-module, PAM = Position-attention-Module.

7.2.2 Model training

The G and D networks are alternately trained by back-propagation in an adversarial fashion: we first fix G and train D for one step using gradients computed from the loss function, and then fix D and train G for another step using gradients computed from the same loss function passed from D to G. Assume x is a skin lesion image containing a lesion, y is the ground-truth of the segmented image of that lesion, and $G(x, z)$ and $D(x, G(x, z))$ are the outputs of the generator and the discriminator, respectively. The generator loss function G comprises three terms: binary cross entropy loss, L_1 norm to boost the outliers, and Jaccard loss to increase the intersection:

$$\begin{aligned} \ell_{Gen}(G, D) = \mathbb{E}_{x,y,z}(-\log(D(x, G(x, z))))+ \\ \lambda\mathbb{E}_{x,y,z}(\ell_{L_1}(y, G(x, z))) + \alpha\mathbb{E}_{x,y,z}(\ell_{Jaccard\,loss}(y, G(x, z))), \end{aligned} \tag{7.7}$$

where λ and α are empirical weighting factors. The variable z is a random variable introduced as a dropout in the decoding layers at both training and testing phases, which helps to generalize the learning process and avoid overfitting. The L_1 *loss* is also necessary to boost the learning process that may be too slow because the adversarial loss term may not properly formulate the gradient towards the expected segmented lesion shape.

In addition, we consider the optimization of the *Jaccard loss* (JL) for the lesion classes. Let G be the hand drawn ground truth of the lession region, and P its respective computer-generated segmentation mask, then binary Jaccard loss that is based on Jaccard distance is defined as follows (Yuan, 2017):

$$d_J(G, P) = 1 - J(G, P) = 1 - \frac{(G \cap P)}{|G| + |P| - |G \cap P|} \tag{7.8}$$

A non-differentiable function $d_J(G, P)$ can be introduced for loss minimization; however, it is not easy to directly apply such function for back-propagation. Moreover, it would be computationally expensive to generate a binary mask from continuous MobileGAN output for each iteration during optimization. Thus, we use

the following loss function:

$$L_{d_J} = 1 - \frac{\sum_{x,y}(g_{xy}, p_{xy})}{\sum_{x,y} g_{xy}^2 + \sum_{x,y} p_{xy}^2 - \sum_{x,y}(g_{xy}p_{xy})} \tag{7.9}$$

Usually in the loss function a weight map is needed to balance the pixels from lession regions and background; however, it is not the case for the above loss. Meanwhile, the Jaccard loss function is differentiable:

$$\begin{aligned} JL = \frac{\delta L_{d_J}}{\delta L_{p_{xy}}} = &-\frac{g_{xy}[\sum_{x,y} g_{xy}^2 + \sum_{x,y} p_{xy}^2 - \sum_{x,y}(g_{xy}p_{xy})]}{[\sum_{x,y} g_{xy}^2 + \sum_{x,y} p_{xy}^2 - \sum_{x,y}(g_{xy}p_{xy})]^2} \\ &+\frac{(2p_{x,y} - g_{xy})[\sum_{x,y}(g_{xy}p_{xy})]}{[\sum_{x,y} g_{xy}^2 + \sum_{x,y} p_{xy}^2 - \sum_{x,y}(g_{xy}p_{xy})]^2} \end{aligned} \tag{7.10}$$

This loss can be efficiently integrated into the backpropagation during network training. If the generator network is optimized properly, the values of $D(x, G(x, z))$ approach 1.0, meaning that the discriminator cannot differentiate the generated segmentation mask from the ground truth, while L_1 and Jaccard losses should approach to 0.0, indicating that every generated mask matches the corresponding ground truth mask both in overall pixel-to-pixel distances (L_1) and in tight convex surrogate (JL) to all Intersection-Over-Union (IoU).

The discriminator loss function D can be formulated as follows:

$$\ell_{Dis}(G, D) = \mathbb{E}_{x,y,z}(-\log(D(x, y))) + \mathbb{E}_{x,y,z}(-\log(1 - D(x, G(x, z)))). \tag{7.11}$$

The optimizer should fit D to maximize the loss values for ground truth images (by minimizing $-\log(D(x, y))$) and to minimize the loss values for the predicted image (by minimizing $-\log(1 - D(x, G(x, z)))$. These two terms compute the binary cross entropy (BCE) loss using both images, assuming that the expected class for ground truth and generated images is 1 and 0, respectively.

7.3 Experiments

Datasets: The efficacy of the proposed model is assessed on two publicly available benchmark datasets of dermoscopic images for skin lesion analysis: ISIC 2018 (Skin Lesion Analysis Towards Melanoma Detection, grand challenge datasets) (Codella et al., 2019) and ISBI 2017 (IEEE International Symposium on Biomedical Imaging, ISBI 2017, grand challenge datasets) (Codella et al., 2018). The ISIC 2018 dataset includes 2,594 images with the corresponding ground truth masks annotated by expert dermatologists. The validation and testing sets contain 100 and 1,000 images, respectively, without ground truth (evaluated by online (ISIC, 2018) only). In our experiments, we used 80% of the training set of the ISIC 2018 dataset for training and 20% for validation as proposed in (Al-Masni et al., 2018). In turn, ISBI 2017 dataset was divided into training, validation and testing sets with 2000, 150 and 600 images, respectively. Note that we trained our model with ISIC 2018 training set and evaluated our model on ISBI 2017 test and ISIC 2018 validation sets.

Evaluation Metrics: Five evaluation metrics are used for assessing the performance of our model, with the ISBI 2017 test dataset: Jaccard index (JAC), Dice coefficient (DIC) and Accuracy (ACC), Specificity (SPE), Sensitivity (SEN) (Codella et al., 2018). We used the threshold Jaccard index JAC_{th} to evaluate our model with ISIC 2018 validation dataset. We used the evaluation metrics of ISBI 2017 challenges for evaluating the segmentation performances: Jaccard index (JAC), Dice coefficient (DIC) and Accuracy (ACC), Specificity (SPE), and Sensitivity (SEN).

Assume A is the ground truth and B is the segmented region (using a segmentation model). The true positive (TP) rate is defined as $TP = A \cap B$, which is the area of the segmented region common in both A and B. The false positive (FP) rate is defined as $\overline{A} \cap B$, which is the segmented area not belonging to A. Similarly, the false negative (FN) rate is defined as $A \cap \overline{B}$, which is the true area missed by the segmentation model. Below, we present the mathematical expressions of the five metrics:

$$ACC = \frac{TP + TN}{TP + TN + TN + FP} \tag{7.12}$$

$$DIC = \frac{2.TP}{2.TP + FP + FN} \tag{7.13}$$

$$JAC = \frac{TP}{TP + FP + FN} \tag{7.14}$$

$$SEN = \frac{TP}{TP + FN} \tag{7.15}$$

$$SPE = \frac{TN}{TN + FP} \tag{7.16}$$

The predicted lesion masks of the ISIC 2018 challenge is scored using a threshold Jaccard index JAC_{th}. First, the Jaccard index for each image is calculated using a pixel-wise comparison of each predicted segmentation with its corresponding ground truth mask. Then, the final score for each image is used as threshold of the Jaccard index (JAC_{th}) as follows,

$$JAC_{th} = \begin{cases} JAC, & \text{if } JAC > 0.650, \\ otherwise. \end{cases} \tag{7.17}$$

Data augmentation: To achieve accurate segmentation results, we augment the two datasets by flipping the images horizontally and vertically, applying gamma reconstruction and changing the contrast using adaptive histogram equalization (CLAHE) with different values on the original RGB images

Implementation: We used Adam (Kingma and Ba, 2014). We achieved the best results with Adam optimizer with parameters $\beta_1 = 0.5$, $\beta_2 = 0.999$. In turn, the learning rate was set to 0.0002 with a batch size of 8. The weighting factors of Jaccard loss and L_1-norm loss (λ and α) were set to 0.1 and 0.5, respectively. Our experiments are carried on NVIDIA 1080Ti with 11GB memory taking around 8 hours to train the network. The model is implemented on PyTorch (Paszke et al., 2017) deep learning library.

Experimental results: The size of the images ranges from 542×718 to 2848×4288 pixels that is considered a very large to train the proposed model. Each input

Table 7.2: Evaluating the proposed model on the ISBI 2017 test dataset

Methods	ACC	DIC	JAC	SEN	SPE	Parameters (million)
FCN (Long et al., 2015)	92.72	83.83	72.17	79.98	96.66	134.3
U-Net (Ronneberger et al., 2015)	90.14	76.27	61.64	67.15	97.24	12.3
SegNet (Badrinarayanan et al., 2017)	91.76	82.09	69.63	80.05	95.37	11.5
FrCN (Al-Masni et al., 2018)	94.03	87.08	77.11	**85.40**	96.69	16.3
SLSDeep (Sarker et al., 2018b)	93.6	**87.8**	78.2	81.6	98.3	46.65
SegAN (Xue et al., 2018)	94.1	86.7	**78.5**	-	-	382.17
Proposed	**97.61**	87.63	77.98	78.50	**99.92**	**2.35**

image was resized to $q \times q$ pixels to speed up the training process of our model. We trained and tested our model with different image sizes (64×64, 128×128 and 256×256). The best segmentation results are obtained with the input size of 128×128 (for detailed results, see the section experimental results for different variations of the proposed MobileGAN model).

Quantitative results of the proposed model on ISBI 2017 test and ISIC 2018 validation sets shows in Table 7.2 and Table 7.3. With ISBI 2017 test dataset, we compared the MobileGAN with five skin lesion segmentation methods (FCN (Long et al., 2015), U-Net (Ronneberger et al., 2015), SegNet (Badrinarayanan et al., 2017), FrCN (Al-Masni et al., 2018), SLSDeep (Sarker et al., 2018b) and an adversarial network, SegAN (Xue et al., 2018)). We took all the test results of FCN, U-Net, SegNet, FrCN from the literature (Al-Masni et al., 2018) that used the same dataset. As shown, the proposed MobileGAN model yields the best results in terms of ACC and SPE. MobileGAN achieves an improvement of the ACC score of 3.51% more top than the SegAN model, and the SPE score of 1.62% higher than the SLSDeep model. In turn, the SLSDeep model yields a little bit better results with an improvement of 0.17% compared to our model. In turn, the SegAN model gives a better JAC score than our model with an improvement of 0.52%. Also, the FrCN model achieves an increase of 6.9% of the SEN score higher than our model. Regarding the ISIC 2018 validation dataset, we compared MobileGAN to the FCN, U-Net, SegNet, FrCN and GAN-FCN models as shown in Table 7.3. We used the validation evaluation of FCN, U-Net, SegNet, FrCN from the literature (Al-masni et al., 2018). Our model achieves the highest JAC_{th} score compared to the GAN-FCN models with an improvement of 0.6% and better than the U-Net model with an increase of 26%.

Table 7.3: Evaluating the proposed model on the ISIC 2018 validation dataset

Methods	JAC_{th}	**Parameters (million)**
FCN (Long et al., 2015)	74.70	134.3
U-Net (Ronneberger et al., 2015)	54.4	12.3
SegNet (Badrinarayanan et al., 2017)	69.50	11.5
FrCN (Al-Masni et al., 2018)	74.60	16.3
GAN-FCN (Bi et al., 2018)	77.80	10.61
Proposed	**78.4**	**2.35**

In addition, we compared the MobileGAN model to the FCN, U-Net, SegNet, FrCN, SegAN and GAN-FCN models in terms of the number of the parameters. The MobileGAN has only 2.35 millions of parameters. While the closest one is the GAN-FCN model with 10.61 millions of parameters. In turn, the SegAN is the most massive model with 382.17 millions of parameters. That model used the traditional GAN model. It is evident that adding non-bottleneck and position and channel attention modules significantly reduced the number of parameters of the MobileGAN model. Besides, Mobile GAN has a number of parameters 57x,5x, 4x, 6x, and 19x lower than the FCN, U-Net, SegNet, FrCN, and SLSDeep models, respectively.

Figure 7.5 shows qualitative segmentation results of the MobileGAN model with some examples from the ISBI 2017 test dataset. As shown, in Figure 7.5 (left), Although the tested images have a high similarity between the color of the lesion and the skin regions, fuzzy boundaries and even very small lesions, the MobileGAN

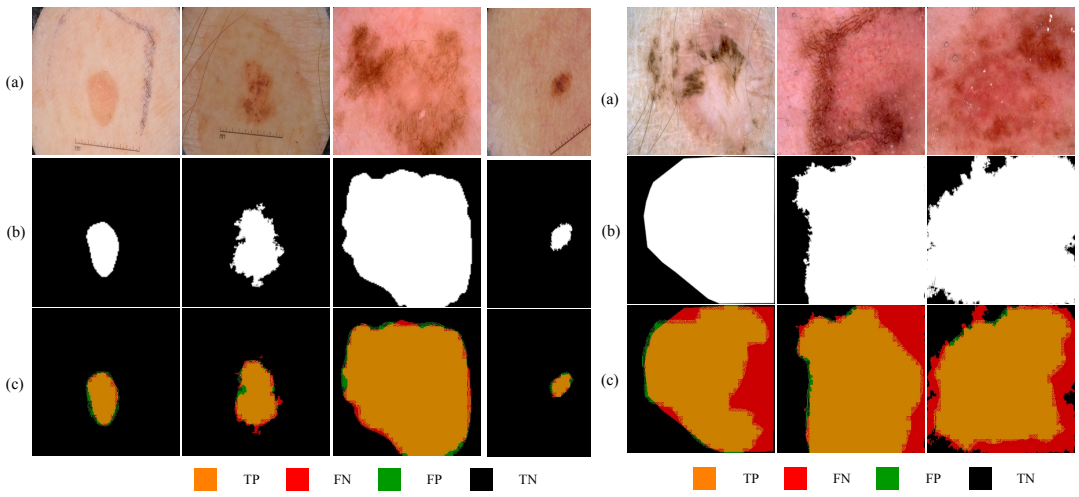


Figure 7.5: Segmentation results of our model: (a) input image (b) ground truth (c) **left:** accurately segmented lesions (c) **right:** incorrectly segmented lesions

model accurately segments the boundary of each skin lesion with an accuracy of about 95%. Besides, in Figure 7.5(right), the four images shown have skin regions (the background) are very small compared to lesion regions, also the lesion regions occupy most of the image and intersect three margins of the images. In these cases, our MobileGAN yields inaccurately segmentation. It is a bit difficult to segment the boundaries of tumors accurately. That means our model needs to a complete shape of the lesion area to properly segment the boundaries of the legions regions.

Experimental results for different variations of the proposed MobileGAN model: In proposed MobileGAN model, we first evaluated a baseline (BL) segmentation model, in which sequential stacking factorized kernels are used on all convolution and deconvolution layers of both encoder and decoder parts of the generator network. In a variation, we tested BL with position attention module (PAM) after every downsampling layer in encoder and upsampling layer in decoder. In another variation, we tested BL+PAM with channel attention module (CAM) after every factorized kernels blocks in encoder and decoder and PAM after the first and CAM after the second downsample block in discriminator network. We assessed all the variation on the ISBI 2017 test dataset. The results are shown in Table 7.4. We found that BL+PAM+CAM is giving the best performance in all metrics with only few number of parameters. We have also done another experiment by adding a multiscale block to BL+PAM+CAM, which we called MobileGAN model. Then we compared BL+PAM+CAM (MobileGAN -multiscale) with the proposed MobileGAN model with input image of 128×128. Table 7.5 shows the comparison of variations of with and without multiscale.

Table 7.4: Evaluating the variations of proposed model on the ISBI 2017 test dataset

Methods	ACC	DIC	JAC	SEN	SPE
BL	96.63	81.61	68.93	69.42	99.91
BL+PAM	96.82	85.03	73.96	**83.72**	98.40
BL+PAM+CAM	**97.61**	**87.63**	**77.98**	78.50	**99.92**

Table 7.5: Evaluating the variations of proposed model on the ISBI 2017 test dataset

Methods	ACC	DIC	JAC	SEN	SPE
MobileGAN-multiscale	96.69	82.58	70.33	72.74	99.59
MobileGAN	**97.61**	**87.63**	**77.98**	**78.50**	**99.92**

Table 7.6: Evaluating the variations of proposed model on the ISBI 2017 test dataset

Input size	ACC	DIC	JAC	SEN	SPE
64x64	93.44	75.59	60.76	**94.16**	93.36
256x256	96.72	84.72	73.49	84.36	98.21
128x128	**97.61**	**87.63**	**77.98**	78.50	**99.92**

Table 7.7: Evaluating the variations of proposed model on the ISBI 2017 test dataset

Loss	ACC	DIC	JAC	SEN	SPE
MobileGAN+BCE+L1	96.90	84.26	72.80	77.05	99.30
MobileGAN+BCE+L1+ Jaccard Loss	**97.61**	**87.63**	**77.98**	**78.50**	**99.92**

We have tested the proposed model on different variations input sizes to see which combination suites the proposed MobileGAN the best. Table 7.6 shows MobileGAN results on 64×64, 128×128 and 256×256, which concludes that it works the best on the image size of 128×128.Finally, we assessed the model on various loss functions to see which loss function suits the proposed model the best. Table 7.7 shows the comparison of the such variations. Fig 7.6 presents test set examples segmented by the proposed model. We can observe that there are TP which has accurately segment the tumor lesion of different shapes.

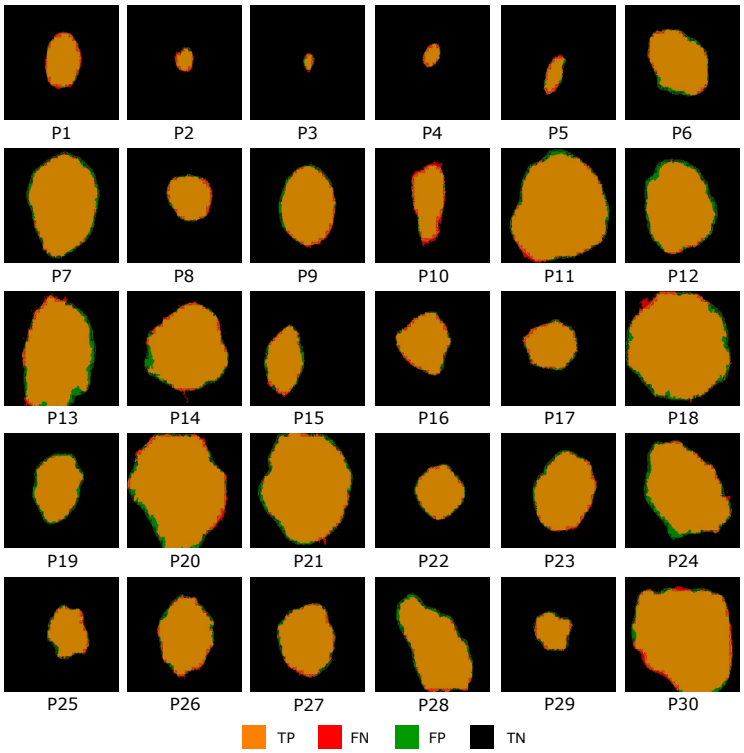


Figure 7.6: The segmentation examples of proposed model on the test set of ISBI 2017.

7.4 Conclusions

In this work, we have proposed a lightweight yet efficient GAN model (MobileGAN) for skin lesion segmentation. The MobileGAN is built by adapting the GAN model by adding 1D non-bottleneck factorization networks with position and channel attention blocks. In comparison to state-of-art skin melanoma segmentation, the number of parameters of MobileGAN model is significantly reduced with only 2.35 millions of parameters. The MobileGAN model has been evaluated on ISBI 2017 test and ISIC 2018 validation datasets. With the ISBI 2017 test dataset, it yields appropriate segmentation results with an accuracy of 97.61%, a specificity of 99.92%. The proposed model also provides Jaccard and sensitivity of 77.98% 78.50%, respectively that is comparable to the state-of-the-art. The proposed model achieves a threshold Jaccard score of 78.4% with the ISIC 2018 validation dataset. Future work attempts to implement a mobile application based on the MobileGAN model to segment skin lesions in images captured by a low-resolution camera.

Chapter 8

Conclusions and Future Work

In this thesis, we designed and evaluated many efficient deep models including, multi-scale Atrous convolutional networks, "MACNet", MACNet with self-attention mechanism, "MACNet+SA" for food environment and "CuisineNet" for food attributes classification; SLSDeep and MobileGAN for skin lesion segmentation.

Initially, fine-tuned baseline models are applied for recognition of food-related scenes in conventional images to identify, *where we eat?*. Transfer learning with Inception-V3 yielded a classification rate of 75.22% among the state-of-the-art models on "FoodPlaces" dataset. Afterwards, we presented "MACNet", based on multi-scale Atrous convolution networks to extract the key features related to food places of the input egocentric images. Here, we evaluated image-level analysis and our proposed model archived comparable performances better than three common architectures of classification methods, namely VGG-16, ResNet-50 and Inception-V3. Eventually, we demonstrated the "MACNet+SA", which is successive of our previous model with attention mechanisms. As we discovered that the frames captured by the wearable camera has the time dependency. Thus, we used self-attention mechanism that can efficiently calculate the attention scores based on a self-representation of every frame. We evaluated event-level analysis and our model archived state-of-the-art performance in term of F1 score of 86% and 80% on validation and test set of "EgoFoodPlaces" dataset, respectively. Finally, we focused on to classify food attributes to understand, *what we eat?*. The aggregation of multi-scale convolution

layers with different kernel sizes, "CuisineNet", is used for weighting the features resulted from different scales. In addition, a joint loss function based on NLL is used to fit the model probability to multi labelled classes for multi-modal classification (cuisine and flavours) task. The proposed model yields 65% and 62% average F_1 score on validation and test set of "Yummly48K", respectively, which outperformed the state-of-the-art models.

On the other hand, we present two robust deep models for skin lesion segmentation tasks. Initially, we introduced "SLSDeep", an encoder-decoder network that is constructed by dilated residual layers, in turn, a pyramid pooling network followed by three convolution layers is used for the decoder. A new loss function was proposed by combining both Negative Log-Likelihood (NLL) and End Point Error (EPE) to accurately segment the boundaries of melanoma regions. The robustness of the proposed model was evaluated on two public databases: ISBI 2016 and ISBI 2017, where the proposed model outperformed the state-of-the-art methods in terms of the segmentation accuracy in both the datasets. Later, we designed a lightweight and efficient GAN model, called "MobileGAN", which combines 1D non-bottleneck factorization networks with position and channel attention modules in a GAN model. The proposed model was evaluated on skin lesion segmentation benchmarks dataset and obtains comparable performance with an accuracy of 97.61%. The proposed network has only 2.35 millions of parameters, which is 57x,5x, 4x, 6x, and 19x lower than FCN, U-Net, SegNet, FrCN, and SLSDeep (sate-of-the-art models for skin lesion segmentation) models, respectively.

Although we presented efficient and accurate deep models for recognition of food-related scene and attributes, there are still some problems to overcome in order to develop an automatic nutrition monitoring system. For example, food places classification is still challenging due to a wide variety of food places environments in real-world, and the wide range of possibilities of how a scene can be captured from the first person's point of view. Collecting images from the real world, cleaning and labelling it needs huge effort with respect to both time and costs. Future work aims to address all these issues, finding the optimal solution and developing

a mobile application based on our proposed efficient deep model that combines an egocentric camera with a personal mobile device to build a dietary record to keep a track on the first-person eating behaviour or routine for following a healthy diet. Another, segmentation of skin lesion in dermoscopic images is a challenge due to their blurry and irregular boundaries. To address this problem, our models already have state-of-the-art performance. In future, a mobile-based skin lesion segmentation tool will be developed that can help dermatologists to segment skin lesions accurately in images captured by a low-resolution camera.

References

Ackley, D. H., Hinton, G. E., and Sejnowski, T. J. (1985). A learning algorithm for boltzmann machines. *Cognitive science*, 9(1):147–169.

Aghaei, M., Dimiccoli, M., Ferrer, C. C., and Radeva, P. (2018). Towards social pattern characterization in egocentric photo-streams. *Computer Vision and Image Understanding*.

Aghaei, M., Dimiccoli, M., and Radeva, P. (2015). Towards social interaction detection in egocentric photo-streams. In *Eighth International Conference on Machine Vision (ICMV 2015)*, volume 9875, page 987514. International Society for Optics and Photonics.

Aguilar, E., Bolaños, M., and Radeva, P. (2017). Food recognition using fusion of classifiers based on cnns. In *International Conference on Image Analysis and Processing*, pages 213–224. Springer.

Akkus, Z., Galimzianova, A., Hoogi, A., Rubin, D. L., and Erickson, B. J. (2017). Deep learning for brain mri segmentation: state of the art and future directions. *Journal of digital imaging*, 30(4):449–459.

Al-masni, M., Al-antari, M., Rivera, P., Valarezo, et al. (2018). Automatic skin lesion boundary segmentation using deep learning convolutional networks with weighted cross entropy. ISIC2018: Skin Image Analysis Workshop and Challenge.

Al-Masni, M. A., Al-antari, M. A., Choi, M.-T., Han, S.-M., and Kim, T.-S.

(2018). Skin lesion segmentation in dermoscopy images via deep full resolution convolutional networks. *Computer methods and programs in biomedicine*, 162:221–231.

Allen, K. (2018). A blueprint to beat cancer. https://www.wcrf.org/int/blog/articles/2018/05/blueprint-beat-cancer.

Apalla, Z., Nashan, D., Weller, R. B., and Castellsague, X. (2017). Skin cancer: epidemiology, disease burden, pathophysiology, diagnosis, and therapeutic approaches. *Dermatology and therapy*, 7(1):5–19.

Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223.

Atlanta, G. (2011). American cancer society; 2011. *American Cancer Society: Cancer Facts and Figures*.

Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495.

Baker, S., Scharstein, D., Lewis, J., Roth, S., Black, M. J., and Szeliski, R. (2011). A database and evaluation methodology for optical flow. *IJCV*, 92(1):1–31.

Beijbom, O., Joshi, N., Morris, D., Saponas, S., and Khullar, S. (2015). Menu-match: Restaurant-specific food logging from images. In *Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on*, pages 844–851. IEEE.

Bengio, Y., Simard, P., Frasconi, P., et al. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166.

Berseth, M. (2017). Isic 2017-skin lesion analysis towards melanoma detection. *arXiv preprint arXiv:1703.00523*.

Bi, L., Feng, D., and Kim, J. (2018). Improving automatic skin lesion segmentation

using adversarial learning based data augmentation.

Bi, L., Kim, J., Ahn, E., and Feng, D. (2017). Automatic skin lesion analysis using large-scale dermoscopy images and deep residual networks. *preprint arXiv:1703.04197*.

Bolanos, M., Dimiccoli, M., and Radeva, P. (2017). Toward storytelling from visual lifelogging: An overview. *IEEE Transactions on Human-Machine Systems*, 47(1):77–90.

Bolaños, M., Ferrà, A., and Radeva, P. (2017). Food ingredients recognition through multi-label learning. In *International Conference on Image Analysis and Processing*, pages 394–402. Springer.

Bolanos, M., Mestre, R., Talavera, E., Giró-i Nieto, X., and Radeva, P. (2015). Visual summary of egocentric photostreams by representative keyframes. In *Multimedia & Expo Workshops (ICMEW), 2015 IEEE International Conference on*, pages 1–6. IEEE.

Bolaños, M. and Radeva, P. (2016). Simultaneous Food Localization and Recognition. *Conference: International Conference on Pattern Recognition*, page 376.

Bolanos, M. and Radeva, P. (2016). Simultaneous food localization and recognition. In *Pattern Recognition (ICPR), 2016 23rd International Conference on*, pages 3140–3145. IEEE.

Bossard, L., Guillaumin, M., and Van Gool, L. (2014). Food-101–mining discriminative components with random forests. In *European Conference on Computer Vision*, pages 446–461. Springer.

Boureau, Y.-L., Ponce, J., and LeCun, Y. (2010). A theoretical analysis of feature pooling in visual recognition. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 111–118.

Bouvrie, J. (2006). Notes on convolutional neural networks.

Cao, L. and Fei-Fei, L. (2007). Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE.

Celebi, M. E., Wen, Q., Iyatomi, H., Shimizu, K., Zhou, H., and Schaefer, G. (2015). A state-of-the-art survey on lesion border detection in dermoscopy images. *Dermoscopy image analysis*, pages 97–129.

Chen, J., Pang, L., and Ngo, C.-W. (2017a). Cross-modal recipe retrieval: How to cook this dish? In *International Conference on Multimedia Modeling*, pages 588–600. Springer.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2016a). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *arXiv preprint arXiv:1606.00915*.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2018). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848.

Chen, L.-C., Papandreou, G., Schroff, F., and Adam, H. (2017b). Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*.

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. (2016b). Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180.

Chollet, F. et al. (2015). Keras: Deep learning library for theano and tensorflow. *URL: https://keras. io/k*, 7(8):T1.

Codella, N., Rotemberg, V., Tschandl, P., Celebi, M. E., et al. (2019). Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic).

Codella, N. C., Gutman, D., Celebi, M. E., Helba, B., Marchetti, M. A., Dusza, S. W., Kalloo, A., Liopyris, K., Mishra, N., Kittler, H., et al. (2017). Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1710.05006*.

Codella, N. C., Gutman, D., Celebi, M. E., Helba, B., Marchetti, M. A., Dusza, S. W., Kalloo, A., Liopyris, K., Mishra, N., Kittler, H., et al. (2018). Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic). In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 168–172. IEEE.

Cuschieri, S. and Mamo, J. (2016). Getting to grips with the obesity epidemic in europe. *SAGE open medicine*, 4:2050312116670406.

Day, G. R. and Barbour, R. H. (2000). Automated melanoma diagnosis: where are we at? *Skin Research and Technology*, 6(1):1–5.

de Wijk, R. A., Polet, I. A., Boek, W., Coenraad, S., and Bult, J. H. (2012). Food aroma affects bite size. *Flavour*, 1(1):3.

Dohan, M. and Tan, J. (2011). Lose it! *International Journal of Healthcare Information Systems and Informatics (IJHISI)*, 6(2):60–65.

e Silva, B. V. R., Rad, M. G., Cui, J., McCabe, M., and Pan, K. (2018). A mobile-based diet monitoring system for obesity management. *Journal of health & medical informatics*, 9(2).

Elfiky, N. M., Khan, F. S., Van De Weijer, J., and Gonzalez, J. (2012). Discriminative compact pyramids for object and scene recognition. *Pattern Recognition*, 45(4):1627–1636.

Esteva, A., Kuprel, B., Novoa, R. A., Ko, et al. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115.

Farinella, G. M., Moltisanti, M., and Battiato, S. (2014). Classifying food images represented as bag of textons. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 5212–5216. IEEE.

Finkelstein, E. A., Trogdon, J. G., Cohen, J. W., and Dietz, W. (2009). Annual medical spending attributable to obesity: payer-and service-specific estimates. *Health affairs*, 28(5):w822–w831.

Fu, J., Liu, J., Tian, H., Fang, Z., and Lu, H. (2018). Dual attention network for scene segmentation. *arXiv preprint arXiv:1809.02983*.

Fukushima, K. (1988). Neocognitron: A hierarchical neural network capable of visual pattern recognition. *Neural networks*, 1(2):119–130.

Furnari, A., Farinella, G., and Battiato, S. (2017). Recognizing Personal Locations From Egocentric Videos. *IEEE Transactions on Human-Machine Systems*, 47(1):1–13.

Furnari, A., Farinella, G. M., and Battiato, S. (2016). Temporal segmentation of egocentric videos to highlight personal locations of interest. pages 474–489.

Gilland, D. (2014). Yummly.py is a python api client for the yummly recipe api. https://github.com/dgilland/yummly.py.

Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Graham, B. (2014). Fractional max-pooling. *arXiv preprint arXiv:1412.6071*.

Greenspan, H., Van Ginneken, B., and Summers, R. M. (2016). Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique. *IEEE Transactions on Medical Imaging*, 35(5):1153–1159.

Grimm, E. R. and Steinle, N. I. (2011). Genetics of eating behavior: established and emerging concepts. *Nutrition reviews*, 69(1):52–60.

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., et al. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, 77:354–377.

Gurrin, C., Albatal, R., Joho, H., and Ishii, K. (2014). A privacy by design approach to lifelogging.

Gutman, D., Codella, N. C., Celebi, E., Helba, B., Marchetti, M., Mishra, N., and Halpern, A. (2016). Skin lesion analysis toward melanoma detection: A challenge at the international symposium on biomedical imaging (isbi) 2016, hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1605.01397*.

Hales, C. M., Fryar, C. D., Carroll, M. D., Freedman, D. S., and Ogden, C. L. (2018). Trends in obesity and severe obesity prevalence in us youth and adults by sex and age, 2007-2008 to 2015-2016. *Jama*, 319(16):1723–1725.

Han, Z., Wei, B., Zheng, Y., Yin, Y., Li, K., and Li, S. (2017). Breast cancer multi-classification from histopathological images with structured deep learning model. *Scientific reports*, 7(1):4172.

He, K., Zhang, X., Ren, S., and Sun, J. (2015a). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034.

He, K., Zhang, X., Ren, S., and Sun, J. (2015b). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence (PAMI)*, 37(9):1904–1916.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Herranz, L., Jiang, S., and Xu, R. (2017). Modeling restaurant context for food recognition. *IEEE Transactions on Multimedia*, 19(2):430–440.

Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Hori, C., Hori, T., Lee, T.-Y., Zhang, Z., Harsham, B., Hershey, J. R., Marks, T. K., and Sumi, K. (2017). Attention-based multimodal fusion for video description. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 4203–4212. IEEE.

Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708.

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456.

ISIC (2018). Isic 2018: Skin lesion analysis towards melanoma detection. https://challenge2018.isic-archive.com/live-leaderboards/.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134.

Jabreel, M., Hassan, F., Abdulwahab, S., and Moreno, A. (2017). Recurrent neural conditional random fields for target identification of tweets. In *CCIA*, pages 66–75.

Jabreel, M., Hassan, F., and Moreno, A. (2018). Target-dependent sentiment analysis

of tweets using bidirectional gated recurrent neural networks. In *Advances in Hybridization of Intelligent Methods*, pages 39–55. Springer.

Jakimovski, G. and Davcev, D. (2019). Using double convolution neural network for lung cancer stage detection. *Applied Sciences*, 9(3):427.

Kardynal, A. and Olszewska, M. (2014). Modern non-invasive diagnostic techniques in the detection of early cutaneous melanoma. *Journal of dermatological case reports*, 8(1):1.

Kemps, E., Tiggemann, M., and Hollitt, S. (2014). Exposure to television food advertising primes food-related cognitions and triggers motivation to eat. *Psychology & health*, 29(10):1192–1205.

Keys, A. (1980). Overweight, obesity, coronary heart disease and mortality. *Nutrition Reviews*, 38(9):297–307.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Koushik, J. and Hayashi, H. (2016). Improving stochastic gradient descent with feedback.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

Lakhani, P. and Sundaram, B. (2017). Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks. *Radiology*, 284(2):574–582.

Lazebnik, S., Schmid, C., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *null*, pages 2169–2178. IEEE.

Le, Q. V., Ngiam, J., Coates, A., Lahiri, A., Prochnow, B., and Ng, A. Y. (2011). On

optimization methods for deep learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pages 265–272. Omnipress.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

LeCun, Y. A., Bottou, L., Orr, G. B., and Müller, K.-R. (2012). Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–48. Springer.

Lee, C.-Y., Gallagher, P. W., and Tu, Z. (2016). Generalizing pooling functions in convolutional neural networks: Mixed, gated, and tree. In *Artificial intelligence and statistics*, pages 464–472.

Li, L.-J., Socher, R., and Fei-Fei, L. (2009). Towards total scene understanding: Classification, annotation and segmentation in an automatic framework. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2036–2043. IEEE.

Li, L.-J., Su, H., Fei-Fei, L., and Xing, E. P. (2010). Object bank: A high-level image representation for scene classification & semantic feature sparsification. In *Advances in neural information processing systems*, pages 1378–1386.

Lin, B. S., Michael, K., Kalra, S., and Tizhoosh, H. (2017a). Skin lesion segmentation: U-nets versus clustering. *arXiv preprint arXiv:1710.01248*.

Lin, G., Milan, A., Shen, C., and Reid, I. (2017b). Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1925–1934.

Lin, M., Chen, Q., and Yan, S. (2013). Network in network. *arXiv preprint arXiv:1312.4400*.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár,

P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Lin, Z., Feng, M., Santos, C. N. d., Yu, M., Xiang, B., Zhou, B., and Bengio, Y. (2017c). A structured self-attentive sentence embedding. *arXiv preprint arXiv:1703.03130*.

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., Van Der Laak, J. A., Van Ginneken, B., and Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88.

Liu, C., Cao, Y., Luo, Y., Chen, G., Vokkarane, V., and Ma, Y. (2016). Deepfood: Deep learning-based food image recognition for computer-aided dietary assessment. In *International Conference on Smart Homes and Health Telematics*, pages 37–48. Springer.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440.

Luo, J. and Boutell, M. (2005). Natural scene classification using overcomplete ica. *Pattern Recognition*, 38(10):1507–1519.

Machinery, C. (1950). Computing machinery and intelligence-am turing. *Mind*, 59(236):433.

Maier, A., Syben, C., Lasser, T., and Riess, C. (2019). A gentle introduction to deep learning in medical image processing. *Zeitschrift für Medizinische Physik*, 29(2):86–101.

Malwade, S., Abdul, S. S., Uddin, M., Nursetyo, A. A., Fernandez-Luque, L., Zhu, X. K., Cilliers, L., Wong, C.-P., Bamidis, P., and Li, Y.-C. J. (2018). Mobile and wearable technologies in healthcare for the ageing population. *Computer methods and programs in biomedicine*, 161:233–237.

Marc'Aurelio Ranzato, F.-J. H., Boureau, Y.-L., and LeCun, Y. (2007). Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *Proc. Computer Vision and Pattern Recognition Conference (CVPR'07). IEEE Press*, volume 127.

Martin, C. (2004). Ovid: Metamorphoses. *New York and London: Norton*.

Matsuda, Y. and Yanai, K. (2012). Multiple-food recognition considering co-occurrence employing manifold ranking. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pages 2017–2020. IEEE.

McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133.

Menabrea, L. F. and Lovelace, A. (1842). Sketch of the analytical engine invented by charles babbage.

Meyers, A., Johnston, N., Rathod, V., Korattikara, A., Gorban, A., Silberman, N., Guadarrama, S., Papandreou, G., Huang, J., and Murphy, K. P. (2015). Im2calories: towards an automated mobile vision food diary. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1233–1241.

Mikołajczyk, A. and Grochowski, M. (2018). Data augmentation for improving deep learning in image classification problem. In *2018 international interdisciplinary PhD workshop (IIPhDW)*, pages 117–122. IEEE.

Minsky, M. and Papert, S. (1969). Perceptrons: Anlntroduction to computational geometry. *MITPress, Cambridge, Massachusetts*.

Mishkin, D. and Matas, J. (2015). All you need is a good init. *arXiv preprint arXiv:1511.06422*.

Moosmann, F., Nowak, E., and Jurie, F. (2008). Randomized clustering forests for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(9):1632–1646.

Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814.

Narrative (2017). The world's most wearable hd video camera - narrative clip 2. http://getnarrative.com/.

Newman, T. (2018). Weight loss reduces skin cancer risk. https://www.medicalnewstoday.com/articles/321901.php.

Noronha, J., Hysen, E., Zhang, H., and Gajos, K. Z. (2011). Platemate: crowdsourcing nutritional analysis from food photographs. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 1–12. ACM.

Nwankpa, C., Ijomah, W., Gachagan, A., and Marshall, S. (2018). Activation functions: Comparison of trends in practice and research for deep learning. *arXiv preprint arXiv:1811.03378*.

Okamoto, K. and Yanai, K. (2016). An automatic calorie estimation system of food images on a smartphone. In *Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management*, pages 63–70. ACM.

Oliva, A. and Torralba, A. (2002). Scene-centered description from spatial envelope properties. In *International Workshop on Biologically Motivated Computer Vision*, pages 263–272. Springer.

Olivas, E. S. (2009). *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques: Algorithms, Methods, and Techniques*. IGI Global.

Oliveira, L., Costa, V., Neves, G., Oliveira, T., Jorge, E., and Lizarraga, M. (2014). A mobile, lightweight, poll-based food identification system. *Pattern Recognition*, 47(5):1941–1952.

Oord, A. v. d., Kalchbrenner, N., and Kavukcuoglu, K. (2016). Pixel recurrent neural networks. *arXiv preprint arXiv:1601.06759*.

Parizi, S. N., Oberlin, J. G., and Felzenszwalb, P. F. (2012). Reconfigurable models for scene recognition. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2775–2782. IEEE.

Paszke, A., Gross, S., Chintala, S., and Chanan, G. (2017). Pytorch.

Peralta, M., Ramos, M., Lipert, A., Martins, J., and Marques, A. (2018). Prevalence and trends of overweight and obesity in older adults from 10 european countries from 2005 to 2013. *Scandinavian journal of public health*, page 1403494818764810.

Qin, J. and Yung, N. H. (2010). Scene categorization via contextual visual words. *Pattern Recognition*, 43(5):1874–1888.

Quattoni, A. and Torralba, A. (2009). Recognizing indoor scenes. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 413–420. IEEE.

Radford, A., Metz, L., and Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.

Rahman, M., Alpaslan, N., and Bhattacharya, P. (2016). Developing a retrieval based diagnostic aid for automated melanoma recognition of dermoscopic images. In *Applied Imagery Pattern Recognition Workshop (AIPR)*, pages 1–7. IEEE.

Ravì, D., Wong, C., Deligianni, F., Berthelot, M., Andreu-Perez, J., Lo, B., and Yang, G.-Z. (2016). Deep learning for health informatics. *IEEE journal of biomedical and health informatics*, 21(1):4–21.

Redbook (2017). The 21 best apps for food journaling. https://www.redbookmag.com/body/healthy-eating/advice/g614/lose-weight-apps-tools.

Research, S. (2017). Weight loss reduces skin cancer risk.

https://www.prnewswire.com/news-releases/deep-learning-in-medical-imaging-a-300m-market-by-2021-300408645.html.

Romera, E., Alvarez, J. M., Bergasa, L. M., and Arroyo, R. (2018). Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 19(1):263–272.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

Rosenblatt, F. (1957). *The perceptron, a perceiving and recognizing automaton Project Para.* Cornell Aeronautical Laboratory.

Rozin, P., Fischler, C., Imada, S., Sarubin, A., and Wrzesniewski, A. (1999). Attitudes to food and the role of food in life in the usa, japan, flemish belgium and france: Possible implications for the diet–health debate. *Appetite*, 33(2):163–180.

Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015a). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015b). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252.

Salvador, A., Hynes, N., Aytar, Y., Marin, J., Ofli, F., Weber, I., and Torralba, A. (2017). Learning cross-modal embeddings for cooking recipes and food images. *Training*, 720:619–508.

Samuel, A. L. (1962). Artificial intelligence: a frontier of automation. *The Annals of the American Academy of Political and Social Science*, 340(1):10–20.

Sánchez, J., Perronnin, F., Mensink, T., and Verbeek, J. (2013). Image classification with the fisher vector: Theory and practice. *International journal of computer vision*, 105(3):222–245.

Sarker, M., Jabreel, M., and Rashwan, H. A. (2018a). Cuisinenet: food attributes classification using multi-scale convolution network. *Artificial Intelligence Research and Development: Current Challenges, New Trends and Applications*, 308:365.

Sarker, M., Kamal, M., Rashwan, H. A., Abdel-Nasser, M., Singh, V. K., Banu, S. F., Akram, F., Chowdhury, F. U., Choudhury, K. A., Chambon, S., et al. (2019a). Mobilegan: Skin lesion segmentation using a lightweight generative adversarial network. *arXiv preprint arXiv:1907.00856*.

SARKER, M. M. K., LEYVA, M., SALEH, A., SINGH, V. K., AKRAM, F., RADEVA, P., and PUIG, D. (2017). Foodplaces: Learning deep features for food related scene understanding. In *Recent Advances in Artificial Intelligence Research and Development: Proceedings of the 20th International Conference of the Catalan Association for Artificial Intelligence, Deltebre, Terres de L'Ebre, Spain, October 25-27, 2017*, volume 300, page 156. IOS Press.

Sarker, M. M. K., Rashwan, H. A., Akram, F., et al. (2018b). Slsdeep: Skin lesion segmentation based on dilated residual and pyramid pooling networks. In *International Conference on MICCAI*, pages 21–29. Springer.

Sarker, M. M. K., Rashwan, H. A., Akram, F., Talavera, E., Banu, S. F., Radeva, P., and Puig, D. (2019b). Recognizing food places in egocentric photo-streams using multi-scale atrous convolutional networks and self-attention mechanism. *IEEE Access*, 7:39069–39082.

Sarker, M. M. K., Rashwan, H. A., Talavera, E., Banu, S. F., Radeva, P., and Puig, D. (2018c). Macnet: Multi-scale atrous convolution networks for food places classification in egocentric photo-streams. In *European Conference on Computer Vision*, pages 423–433. Springer.

Scherer, D., Müller, A., and Behnke, S. (2010). Evaluation of pooling operations in convolutional architectures for object recognition. In *International conference on artificial neural networks*, pages 92–101. Springer.

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61:85–117.

Sebag, A., Schoenauer, M., and Sebag, M. (2017). Stochastic gradient descent: Going as fast as possible but not faster. In *OPTML 2017: 10th NIPS Workshop on Optimization for Machine Learning*.

Siegel, R. L., Miller, K. D., and Jemal, A. (2017). Cancer statistics, 2017. *CA: A Cancer Journal for Clinicians*, 67(1):7–30.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Singh, V. K., Rashwan, H. A., Romani, S., Akram, F., Pandey, N., Sarker, M. M. K., Saleh, A., Arenas, M., Arquez, M., Puig, D., et al. (2019). Breast tumor segmentation and shape classification in mammograms using generative adversarial and convolutional neural network. *Expert Systems with Applications*, page 112855.

Sirinukunwattana, K., e Ahmed Raza, S., Tsang, Y.-W., Snead, D. R., Cree, I. A., and Rajpoot, N. M. (2016). Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images. *IEEE Trans. Med. Imaging*, 35(5):1196–1206.

Sparkes, B. A. (2013). *The red and the black: studies in Greek pottery*. Routledge.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Sudderth, E. B., Torralba, A., Freeman, W. T., and Willsky, A. S. (2005). Learning hierarchical models of scenes, objects, and parts. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1331–1338. IEEE.

Suk, H.-I. and Shen, D. (2016). Deep ensemble sparse regression network for alzheimer's disease diagnosis. In *International Workshop on Machine Learning in Medical Imaging*, pages 113–121. Springer.

Sutskever, I., Martens, J., Dahl, G., and Hinton, G. (2013). On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pages 1139–1147.

Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. A. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

Tandy, D. W. and Neale, W. C. (1996). *Works and Days: a translation and commentary for the social sciences*. Univ of California Press.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Wang, T., Wu, D. J., Coates, A., and Ng, A. Y. (2012). End-to-end text recognition with convolutional neural networks. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pages 3304–3308. IEEE.

Wang, Y., Cang, S., and Yu, H. (2018). A data fusion-based hybrid sensory system for older people's daily activity and daily routine recognition. *IEEE Sensors Journal*, 18(16):6874–6888.

WHO (2018a). Cancer. https://www.who.int/news-room/fact-sheets/detail/cancer.

WHO (2018b). Obesity and overweight. https://www.who.int/news-room/fact-sheets/detail/obesity-and-overweight.

Wu, R., Wang, B., Wang, W., and Yu, Y. (2015). Harvesting discriminative meta objects with deep cnn features for scene classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1287–1295.

Xiao, J., Hays, J., Ehinger, K. A., Oliva, A., and Torralba, A. (2010). Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3485–3492. IEEE.

Xu, B., Wang, N., Chen, T., and Li, M. (2015a). Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y. (2015b). Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057.

Xu, R., Herranz, L., Jiang, S., Wang, S., Song, X., and Jain, R. (2015c). Geolocalized modeling for dish recognition. *IEEE transactions on multimedia*, 17(8):1187–1199.

Xue, Y., Xu, T., and Huang, X. (2018). Adversarial learning with multi-scale loss for skin lesion segmentation. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 859–863. IEEE.

Yang, G.-Z., Andreu-Perez, J., Hu, X., and Thiemjarus, S. (2014). Multi-sensor fusion. In *Body sensor networks*, pages 301–354. Springer.

Yang, S., Chen, M., Pomerleau, D., and Sukthankar, R. (2010). Food recognition using statistics of pairwise local features. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2249–2256. IEEE.

Yoo, D., Park, S., Lee, J.-Y., and So Kweon, I. (2015). Multi-scale pyramid pooling

for deep convolutional representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 71–80.

Yu, F. and Koltun, V. (2015). Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*.

Yu, F., Koltun, V., and Funkhouser, T. (2017a). Dilated residual networks. In *Computer Vision and Pattern Recognition*, volume 1.

Yu, J., Tao, D., Rui, Y., and Cheng, J. (2013). Pairwise constraints based multiview features fusion for scene classification. *Pattern Recognition*, 46(2):483–496.

Yu, L., Chen, H., Dou, Q., Qin, J., and Heng, P.-A. (2017b). Automated melanoma recognition in dermoscopy images via very deep residual networks. *IEEE transactions on medical imaging*, 36(4):994–1004.

Yuan, Y. (2017). Automatic skin lesion segmentation with fully convolutional-deconvolutional networks. *arXiv preprint arXiv:1703.05165*.

Yummly (2009). Yummly: Personalized recipe recommendations and search. http://www.yummly.com.

Zacharaki, E. I., Wang, S., Chawla, S., Soo Yoo, D., Wolf, R., Melhem, E. R., and Davatzikos, C. (2009). Classification of brain tumor type and grade using mri texture and shape in a machine learning scheme. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 62(6):1609–1618.

Zeiler, M. D. (2012). Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Zhang, H., Dana, K., Shi, J., Zhang, Z., Wang, X., Tyagi, A., and Agrawal, A. (2018). Context encoding for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7151–7160.

Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., and Metaxas, D. N.

(2017). Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5907–5915.

Zhang, W., Yu, Q., Siddiquie, B., Divakaran, A., and Sawhney, H. (2015). "snap-n-eat" food recognition and nutrition estimation on a smartphone. *Journal of diabetes science and technology*, 9(3):525–533.

Zhang, X. (2017). Melanoma segmentation based on deep learning. *Computer Assisted Surgery*, 22(sup1):267–277.

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). Pyramid scene parsing network. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2881–2890.

Zheng, L., Wang, S., He, F., and Tian, Q. (2014). Seeing the big picture: Deep embedding with contextual evidences. *arXiv preprint arXiv:1406.0132*.

Zhou, B., Khosla, A., Lapedriza, A., Torralba, A., and Oliva, A. (2016). Places: An image database for deep scene understanding. *arXiv preprint arXiv:1610.02055*.

Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., and Torralba, A. (2018). Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464.

Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., and Oliva, A. (2014). Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495.

Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., and Torralba, A. (2017). Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference CVPR*.

UNIVERSITAT
ROVIRA i VIRGILI