

T.C.

**BAHÇEŞEHİR UNIVERSITY**

**FACULTY OF ENGINEERING AND NATURAL SCIENCES**

**DEPARTMENT OF COMPUTER ENGINEERING**

**PROJECT 0901**

**EXTRACTING FOOD CHARACTERISTICS IN MEALS BASED ON DEEP  
NEURAL NETWORKS: A COMPARATIVE STUDY**

Submitted by

FURKAN KILIÇASLAN 1504164

GÜLNUR HÜLAĞA 1243945

EYYÜP GÜZEL 1616643

Advisor

**ÖVGÜ ÖZTÜRK**

İSTANBUL, May 2019

## **STUDENT DECLARATION**

By submitting this report, as partial fulfillment of the requirements of CMP4999, I promise on penalty of failure of CMP4999 that;

- we have given credit to and declared (by citation), any work that is not our own (e.g. parts of the report that is copied/pasted from the Internet, design or construction performed by another person, etc.);
- we have not received unpermitted aid for the project design, construction, report or presentation;
- we have not falsely assigned credit for work to another student in the group, and not taken credit for work done by another student in the group.

## SUMMARY

Food recognition in still images is a problem that has been recently introduced in computer vision. The benchmark data sets used in training and evaluation of food recognition methods contain sample images of popular foods from the globe. However, when they are examined thoroughly, it can be observed that very few of them are Turkish dishes. In this study, we first carry out a data collection process for Turkish dishes and construct a new dataset named "TurkishFoods-10" containing 250 images in each food class. In addition, we introduce a novel food recognition approach that depends on fine-tuning Google Inception v3 deep neural network model based on transfer learning. For this purpose, our Turkish cuisine dataset was used in training an Inception v3 model and it was compared with the widely used Food-101 dataset from the literature and the performance analysis of the developed deep learning-based approach is carried out on this dataset containing 101 food classes. Our results show that the recognition of Turkish dishes can be achieved with certain success even though it does have certain shortcomings and areas to improve upon.

**Keywords:** Deep learning, machine learning, computer vision, multiclass classification, food recognition

# ÖZET

Hareketsiz görüntülerde gıdanın tanınması, son zamanlarda bilgisayarlı görüşte ortaya çıkan bir problemdir. Gıda tanıma yöntemlerinin eğitimi ve değerlendirilmesinde kullanılan kıyaslama veri setleri dünyadaki popüler gıdaların örnek resimlerini içerir. Ancak, iyice incelendiklerinde, çok azının Türk yemeği olduğu görülmektedir. Bu çalışmada ilk olarak Türk yemekleri için bir veri toplama süreci gerçekleştirdik ve her bir yemek sınıfında 250 görüntü içeren "TurkishFoods-10" adlı yeni bir veri seti oluşturduk. Ek olarak, transfer öğrenmeye dayanarak Google Inception v3 derin sinir ağı modelini kullanıp geliştirerek yeni bir gıda tanıma yaklaşımı sunuyoruz. Bu amaçla, Türk mutfağı veri setimiz literatürdeki yaygın olarak kullanılan 101 farklı yemek sınıfı içeren Food-101 veri seti ile karşılaştırmalı bir şekilde geliştirilen derin öğrenme temelli yaklaşımın performans analizi gerçekleştirildi. Elde ettiğimiz sonuçlara göre, Türk yemeklerinin tanınmasında, belli eksiklikler ve üzerinde çalışılabilir alanlar olsa da bu hedefin başarıya ulaşabileceği gösterilmiştir.

**Anahtar Kelimeler:** Derin öğrenme, makine öğrenmesi, bilgisayarla görme, çoklu sınıf sınıflandırma, yemek tanıma

# TABLE OF CONTENTS

|                                          |     |
|------------------------------------------|-----|
| SUMMARY                                  | II  |
| ÖZET                                     | III |
| TABLE OF CONTENTS                        | IV  |
| LIST of TABLES                           | V   |
| LIST of FIGURES                          | V   |
| LIST of ABBREVIATIONS                    | VI  |
| 1 Introduction 1                         |     |
| 1.1 Description of Project               | 1   |
| 2 Related Work 1                         |     |
| 3 Methods and Design 2                   |     |
| 3.1 Technical Limitations and Challenges | 3   |
| 3.2. Facilities and Components           | 3   |
| 3.3. Product Design                      | 3   |
| 4 Experimental Results 4                 |     |
| 5 Work Plan and Costs 5                  |     |
| 5.1 Deliverables and Division of Tasks   | 5   |
| 5.2 Tasks and Time Line                  | 5   |
| 5.3 Cost                                 | 5   |
| 6 Conclusion 6                           |     |
| 7 References8                            |     |
| 8 Appendix A 9                           |     |
| Xbox 360 Technical Specifications        | 9   |
| 9 Appendix B 11                          |     |



## **LIST of TABLES**

Table 1. Components and their estimated costs.

## **LIST of FIGURES**

|                                                                 |   |
|-----------------------------------------------------------------|---|
| Figure 1. Path of electricity from the power plant to the home. | 2 |
| Figure 2. Overall design of the communication system.           |   |
| Figure 3. Proposed experimental setup to verify specifications. | 4 |

## **LIST of ABBREVIATIONS**

|       |                               |
|-------|-------------------------------|
| DNN   | Deep Neural Network           |
| CNN   | Convolutional Neural Networks |
| TF-10 | TurkishFood10                 |
| SVM   | Support Vector Machines       |

# 1 Introduction

This study aims to provide a deep learning based solution to classification of food images from Turkish cuisine as well as comparing the results to similar applications and research that use different type of foods from the globe.

Tracking the eating habits of individuals has become a crucial factor in helping dieticians. This study creates a foundation by enabling high accuracy food image classification for possible applications of food tracking, calorie counting and digital dietary advice through mobile applications that enable the users to take a picture of their food and receive critical information regarding their diet and the food that they are about to eat. There are many different approaches to this problem such as food classification as well as object localization using YOLO architecture and recipe extraction after traditional classification methods.

In our study, we aimed to create a 10 class Turkish food dataset of 250 images each. Then we performed a multiclass classification using transfer learning and convolutional neural networks. We have performed this with both our own dataset and the well known food-101 dataset. We have obtained over %93 validation accuracy on the TF-10 dataset. Using the same architecture we obtained over %94 validation accuracy on 10 random classes from Food-101 dataset. This is a significant level of accuracy regarding that the TF-10 dataset contains only 1/4th of the number of images there are in Food-101 and the architecture we use is an experimental one with no fine-tuning performed. This study firstly provides a small but useful dataset TF-10 and then shows that it is possible to classify 10 classes of Turkish food with relatively high accuracy which can then be used in further applications to track dietary habits etc.

## 1.1 Description of the Project

The task of classification of food images forms a very essential problem in food multimedia application development. Extracting food type from images provides many benefits for human life, such as diabetics control, nutrition value analysis, efficient health statistics analysis, etc.. Due to this fact, for the last decade, researchers have been working on food image classification with an increasingly growing interest in the last two years due to ground-breaking developments in Machine Learning. Deep Learning Architectures(Deep Neural Networks) have achieved much higher recognition accuracies over conventional SVM-based methods for object recognition.

## 2 Related Work

-Calorie MAMA AI

Food AI image recognition technology utilizes convolutional neural networks and deep learning to identify thousands of food categories.

Food Photo Recognition, Packaged Goods Photo Recognition, Easy Diet tracking, International Food Database, Meal Plans

-Foodai

Smart Food Recognition with state-of-the-art Visual Recognition technology. FoodAI™ offers cutting-edge food image recognition technologies for advancing AI in food and healthcare, particularly for Singapore local food. The food image recognition technology is powered by state-of-the-art deep learning techniques

-Bite AI

Bitesnap, a new food journal app designed around photos. It makes tracking what you eat as easy as taking a picture, and it's launching today for iPhone and Android

Our difference is to recognize Turkish food varieties

Food intake assessment is important for obesity management, which has shown significant impacts on public healthcare. Conventional dietary record based food intake

assessment methods exhibit insufficient popularity due to their low accuracy and high dependence on human interactions. [1]

### 3 Methods and Design

In order to achieve high accuracy in this food recognition task, we have decided to use transfer learning using Google’s Inceptionv3 Architecture. We have curated a Turkish food dataset consisting of 10 classes; “Aşure, Baklava, Bulgur Pilavı, Helva, İmam Bayıldı, Kadayıf, Kadınbudu Köfte, Kebap, Sarma, Şehriye Çorbası”. This dataset contains 250 images of each class and it has 2500 images in total ready for training. Even though this is a good starting point for training a deep learning model the number of images required to get high accuracy levels are pretty high so we have decided to use data augmentation methods to improve our data volume. We have used shearing, zooming and horizontal flipping to triple our data volume. All of the images have been rescaled to 299 by 299 before training.

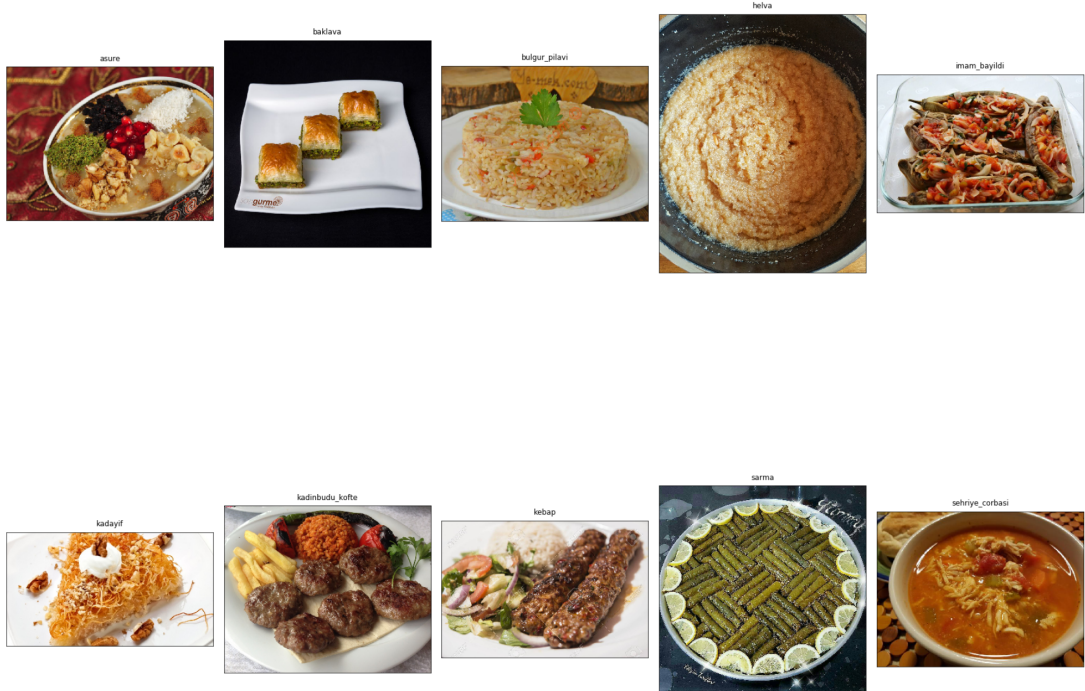


Figure 1. Sample images from the dataset

The inception network used in this project is a complicated architecture built by Google which has outranked many of the state of the art models on benchmark tests.

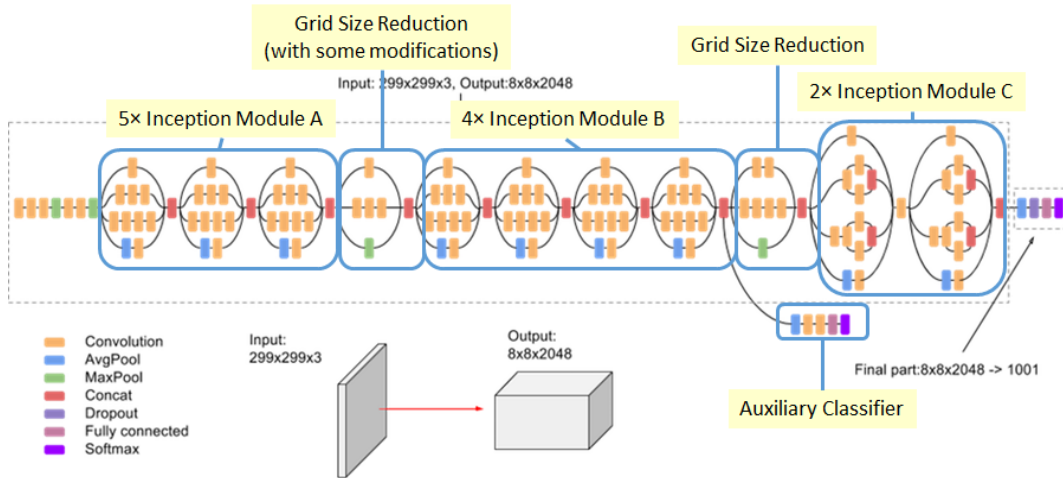


Fig2. Inceptionv3 Architecture; Sik-Ho Tsang 2018 Review: Inception-v3—1st Runner Up (Image Classification) in ILSVRC 2015  
Retrieved from: <https://medium.com/@sh.tsang/review-inception-v3-1st-runner-up-image-classification-in-ilsvrc-2015-17915421f77c>

We have added a GlobalAveragePooling layer and a dense layer on top of the model to train with our data as well as a dropout layer to decrease overfitting issues. Finally, we have added a 10 unit dense layer with softmax activation and l2 regularization to classify the images. The model was trained Stochastic Gradient Descent optimization with categorical cross-entropy loss for 30 epochs and a 80/20 validation split.

### 3.1 Technical Limitations and Challenges

Having such little time and computing resources, as well as low data volume, proved the biggest challenges for us. We had to narrow our study to 10 classes instead of 101 classes as it was virtually impossible to obtain that much data and train such a big neural network with that much data. Also, we first diverted our focus on many different approaches and found this to be distracting as well as diluting our resources. So we pivoted to a narrower application to the end of the project.

## 3.2. Facilities and Components

This project only required a powerful GPU in terms of hardware which thankfully one of the team members already had in possession. For the collection of the dataset, we have used several web scraping methods and manual extraction. The most important tools we have used for the Deep Learning model was Keras, a tensorflow wrapper for building neural network architectures and compiling them. Keras is a very user-friendly framework in contrast to Pytorch, another deep learning framework developed by Facebook which focuses more on flexibility at the expense of user-friendliness. Other than that, utility libraries such as matplotlib and numpy were also used to manipulate data and plot images. All of the codes were run on a jupyter notebook on a local machine with an Nvidia GTX 1050Ti graphics card.

## 3.3. Product Design

The main flow of the application is very simple. The data is processed to be fed into the network. The network starts training for 30 epochs based on all the parameters given earlier. The best model which is the one with the least validation loss is saved and the model is used to predict the food class of any given image.

# 4 Experimental Results

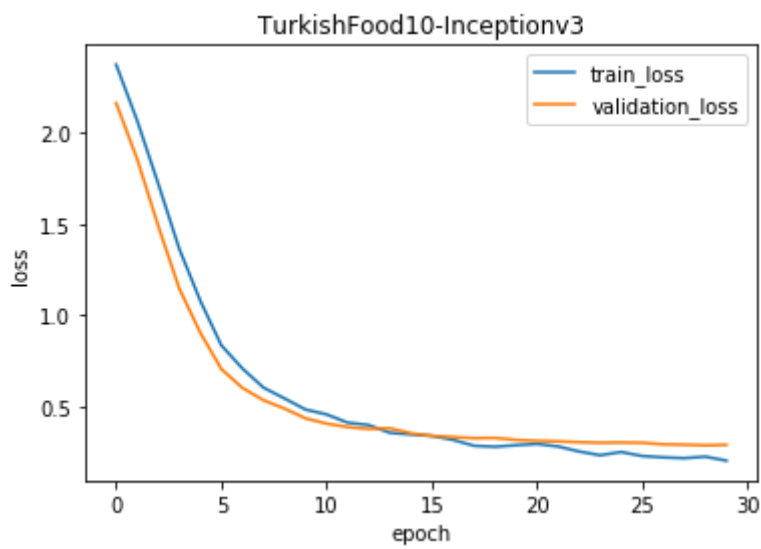
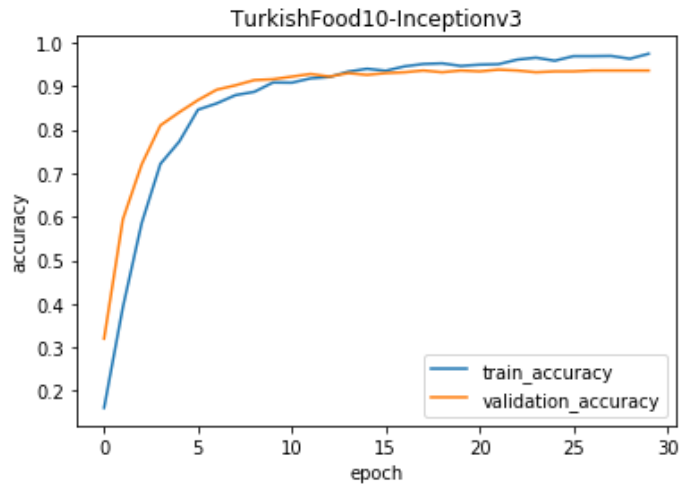
We have trained two different models with the same parameters and hyperparameters. One with 10 random classes from the food-101 dataset consisting of 10.000 images in total for training and validation. The other with the 10 classes of the TurkishFood-10 dataset consisting of 2.500 images in total for training and validation. After 30 epochs of training Food101 model has obtained the following results :

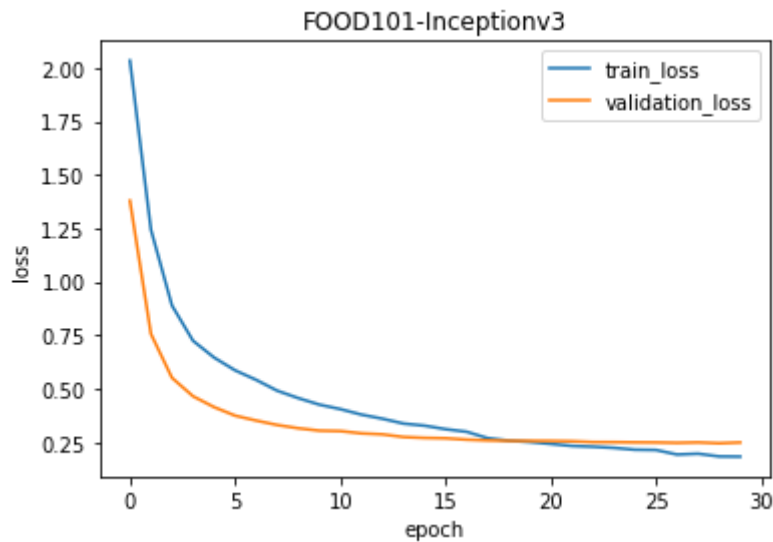
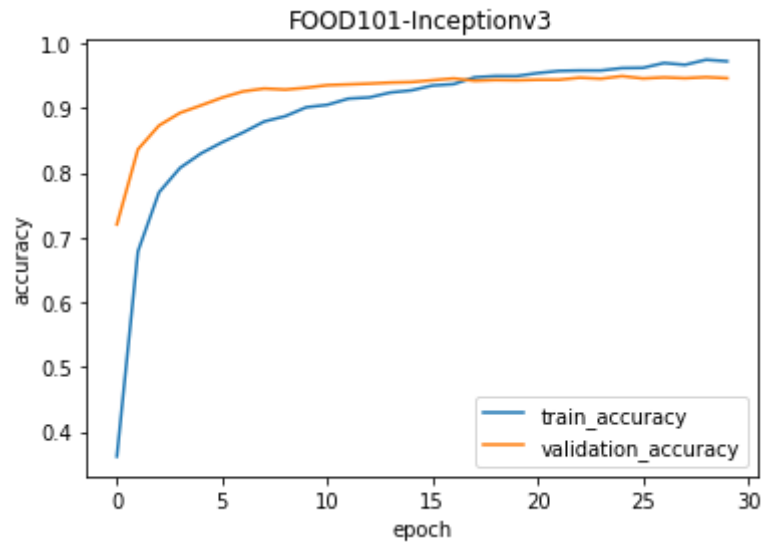
loss: 0.1825 - acc: 0.9725 - val\_loss: 0.2487 - **val\_acc: 0.9464**

Meanwhile, the same model trained with our TurkishFood10 dataset obtained the following results:

loss: 0.2060 - acc: 0.9745 - val\_loss: 0.2922 - **val\_acc: 0.9360**

These results are obtained as a baseline for this study without any hyperparameter tuning due to resource and time limitations. Following are the Accuracy and Loss plots for both of the models.





## 5 Work Plan and Costs

### 5.1 Deliverables and Division of Tasks

1. Research and Design;

2. CMP students made research and plan about the project. They found existing codes for our project. They found data sets for Turkish foods. They organized these data sets to make them suitable for the project. Necessary files were prepared for the data set.

## 2.Construction;

1 CMP student worked on the existing codes. He made codes executable. He worked on faulty codes. He made training using prepared data sets. He ran the codes and took the results.

## 3.Testing and Documentation;

All CMP students tested the project and prepared the documents according to the results.They prepared a report and a presentation.

## 5.2 Tasks and Time Line

### 1.Research and Design;

Eyyüp Güzel and Gülnur Hılağa worked at home, university and laboratory. Researched articles, code and pictures from the internet.

### 2.Construction

Furkan Kılıçaslan worked at home, university and laboratory. He wrote the codes using various programs.

### 3.Testing and Documentation;

Eyyüp Güzel, Gülnur Hılağa and Furkan Kılıçaslan worked together at home, university and laboratory. They tested the project and prepared a report and a presentation.

Table 2. The project Gantt chart.

| TASK LIST                                              | WEEKS |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |
|--------------------------------------------------------|-------|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|
|                                                        | 1     | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
| <b>Research and Design</b>                             |       |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |
| General research about the project.                    |       |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |
| Making a plan and determining the programs to be used. |       |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |
| <b>2. Coding</b>                                       |       |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |

|                                           |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|-------------------------------------------|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|
| Testing and organizing existing code      |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Studying resnet,alexnet and vgg           |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Completing missing data sets and training |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Running and getting results               |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| <b>Testing and Documentation</b>          |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Prepare the proposal                      |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Verification                              |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Prepare the report                        |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Presentation                              |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

## 6 Conclusion

This study realizes that it is possible to obtain a classification accuracy of higher than %93 even with such little data and no fine tuning over transfer learning. These results show promising potential in the classification of food images which is the foundational technology behind automated systems for dietary advice, nutrition analysis, calorie counting etc. This study also curates a small but useful dataset specific for Turkish cuisine. Unfortunately, due to time and resource constraints, we could not delve into these applications.

This study has been a crucial point for our group where we learned the importance of planning ahead, having emergency plans and working together in harmony. Also, it has helped us learn about technical details of deep learning, computer vision and image recognition. We are confident that this project laid a strong foundation for us to keep working on deep learning applications in the future.

## 7 References

- Xin Wang et al. (2015) Recipe Recognition With Large Multimodal Food Dataset
- Qian Yu et al. (2016) Deep Learning Based Food Recognition
- Yuzhen Lu et al. (2019) Food Image Recognition by Using Convolutional Neural Networks (CNNs)
- Binh T. Nguyen et al. (2017) A Deep Learning based Food Recognition System for Lifelog Images
- Bossard et al. (2014) Food-101 -- Mining Discriminative Components with Random Forests
- Sik-Ho Tsang (2018) Review: Inception-v3—1st Runner Up (Image Classification) in ILSVRC 2015 Retrieved [Blog Post] from: <https://medium.com/@sh.tsang/review-inception-v3-1st-runner-up-image-classification-in-ilsvrc-2015-17915421f77c>
- Fatih İyiyol (2010) The Effect of Turkish Culture on Bosnian Culture Exemplified by Food

## Appendix A

- Model Summary of the Modified Inception v3 Architecture (Find as a separate document)

## 8 Appendix B

Code for preparing the data and further training an inception model

```
K.clear_session()
n_classes = 10
img_width, img_height = 299, 299
train_data_dir = src_train
validation_data_dir = src_test
nb_train_samples = 2000 #75750
nb_validation_samples = 500 #25250
batch_size = 16

train_datagen = ImageDataGenerator(
    rescale=1. / 255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True)

test_datagen = ImageDataGenerator(rescale=1. / 255)

train_generator = train_datagen.flow_from_directory(
    train_data_dir,
    target_size=(img_height, img_width),
    batch_size=batch_size,
    class_mode='categorical')

validation_generator = test_datagen.flow_from_directory(
    validation_data_dir,
    target_size=(img_height, img_width),
    batch_size=batch_size,
    class_mode='categorical')

inception = InceptionV3(weights='imagenet', include_top=False)
x = inception.output
x = GlobalAveragePooling2D()(x)
x = Dense(128, activation='relu')(x)
x = Dropout(0.2)(x)

predictions = Dense(10, kernel_regularizer=regularizers.l2(0.005), activation='softmax')(x)

model = Model(inputs=inception.input, outputs=predictions)
```

```
model.compile(optimizer=SGD(lr=0.0001, momentum=0.9), loss='categorical_crossentropy',
metrics=['accuracy'])
checkpointer = ModelCheckpoint(filepath='best_model_10class.hdf5', verbose=1,
save_best_only=True)
csv_logger = CSVLogger('history_10class.log')

history = model.fit_generator(train_generator,
                             steps_per_epoch = nb_train_samples // batch_size,
                             validation_data=validation_generator,
                             validation_steps=nb_validation_samples // batch_size,
                             epochs=30,
                             verbose=1,
                             callbacks=[csv_logger, checkpointer])

model.save('model_trained_10class.hdf5')
```