##### DETECTION OF DEFECTIVE POTATO CHIPS

**A PROJECT REPORT**

###### ***Submitted by***

TEAM 11

**VANSH RAJA**(21BAI10067)

**ANUJ VASTANI** (21BAI10210)

**HARSHIT SINGH** (21BAI10023)

**ARNAV GUPTA**(21BAI10191)

**AVANISH KUMAR SINGH** (21BAI10004)

*in partial fulfillment for the award of the degree*

*of*

##### **BACHELOR OF TECHNOLOGY**

*in*

**COMPUTER SCIENCE AND ENGINEERING**

****

**SCHOOL OF COMPUTING SCIENCE AND ENGINEERING**

**VIT BHOPAL UNIVERSITY**

**KOTHRI KALAN, SEHORE**

**MADHYA PRADESH - 466114**

MAR 2023

**VIT BHOPAL UNIVERSITY, KOTHRI KALAN, SEHORE**

**MADHYA PRADESH – 466114**

**BONAFIDE CERTIFICATE**

Certified that this project report titled **“DETECTION OF DEFECTIVE POTATO CHIPS”** is the bonafide work of “**VANSH RAJA(21BAI10067) ,ANUJ VASTANI**

**(21BAI10210) ,HARSHIT SINGH(21BAI10023) ,ARNAV GUPTA**

**(21BAI10191), AVANISH KUMAR SINGH(21BAI10004)”** who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported at this time does not form part of any other project/research work based on which a degree or award was conferred on an earlier occasion on this or any other candidate.

**PROGRAM CHAIR PROJECT GUIDE**

<<Name>>,<<Designation>> <<Name>>,<< Designation>>

School of Computer Science and Engineering School of Computer Science and Engineering

VIT BHOPAL UNIVERSITY VIT BHOPAL UNIVERSITY

The Project Exhibition II Examination is held on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ACKNOWLEDGEMENT**

First and foremost I would like to thank the Lord Almighty for His presence and immense blessings throughout the project work.

I wish to express my heartfelt gratitude to Dr S Suthir, Head of the Department, School of Computer Science for much of his valuable support and encouragement in carrying out this work.

I would like to thank my internal guide Mr. Nilamadhab Mishra,for continually guiding and actively participating in my project, giving valuable suggestions to complete the project work.

I would like to thank all the technical and teaching staff of the School of Computer Science, who extended directly or indirectly all support.

Last, but not least, I am deeply indebted to my parents who have been the greatest support while I worked day and night for the project to make it a success.

**LIST OF GRAPHS**

| **FIGURE NO.** | **TITLE** | **PAGE NO.** |
| --- | --- | --- |
| **1** | **LOSS VS NO. OF EPOCHS** | **19** |
| **2** | **TRAINING VS VALIDATION ACCURACY** | **19** |
| **3** | **TRAINING VS VALIDATION LOSS** | **20** |

**LIST OF TABLES**

| **TABLE NO.** | **TITLE** | **PAGE NO.** |
| --- | --- | --- |
| **1** | **TABLE1** | **18** |

**ABSTRACT**

Here in our project,

**PURPOSE**: Applying Artificial Intelligence to make a model using a Convolutional Neural Network that classifies a potato chip into defective and non-defective based on the image provided to it.

**METHODOLOGY**:

* Data collection: Collect a dataset of images of potato chips, including both normal and defective chips. This dataset should be large and diverse enough to cover all possible variations of defective chips.
* Data preparation: Preprocess the collected dataset by cleaning, resizing, and normalizing the images. You may also need to label the images to create a ground truth for the classification task.
* Model selection: Choose a suitable machine learning algorithm for classification, such as a convolutional neural network (CNN). CNNs are a popular choice for image classification problems due to their ability to learn hierarchical features from images.
* Model training: Train the selected model on the prepared dataset using appropriate hyperparameters and optimization algorithms. The training process should be monitored for convergence and overfitting.
* Model evaluation: Evaluate the performance of the trained model on a separate validation dataset to assess its accuracy, precision, and recall.
* Model deployment: Once the trained model is sufficiently accurate, deploy it in a real-world setting to automatically detect defective potato chips. This could be done using a camera and computer vision algorithms to process the images in real-time.
* Continuous improvement: Continuously monitor the performance of the deployed model and collect feedback from users to improve its accuracy and reliability.

**FINDINGS:** Potato chip manufacturers spend a lot of time, money and manpower into detection and exclusion of defective potato chips and this whole process could be made a lot more efficient if it was automated with the help of AI.

**TABLE OF CONTENTS**

| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
| --- | --- | --- |
|  | List of Abbreviations  List of Figures and Graphs  List of Tables  Abstract | iii  iv  v  vi |
| 1 | **CHAPTER-1:**  **PROJECT DESCRIPTION AND OUTLINE** Introduction 1.2 Motivation for the work  1.3 Problem Statement  1.4 Organization of the project | 7 |
| 2 | **CHAPTER-2:**  **RELATED WORK INVESTIGATION**  2.1 Introduction  2.2 Neural Networks  2.3 Types of neural networks  2.4 GPU and TPU  2.5 GPU vs TPU | 8 |
| 3 | **CHAPTER-3:**  **REQUIREMENT ARTIFACTS**  3.1 Introduction  3.2 Hardware and Software requirements  3.3 Python Libraries | 15 |
| 4 | **CHAPTER-4:**  **DESIGN METHODOLOGY AND ITS NOVELTY**  4.1 Methodology and Goal  4.2 Modules/Sub-modules imported  4.3 Novelty | 17 |
| 5 | **CHAPTER-5:**  **TECHNICAL IMPLEMENTATION & ANALYSIS**  5.1Technical implementation  5.2 Analysis  5.3 Loss vs No. of Epochs  5.4 Training vs Validation Accuracy  5.5 Training vs Validation Loss | 12 |
| 6 | **CHAPTER-6:**  **PROJECT OUTCOME AND APPLICABILITY**  6.1Key implementation of project | 14 |
| 7 | **CHAPTER-7:**  **CONCLUSIONS AND RECOMMENDATION**  7.1Conclusion  7.2 Recommendation/Bibliography | 15 |
|  | References  ***Note: List of References should be written as per IEEE/Springer reference format. (Specimen attached)*** | 16 |

**CHAPTER-1**

**PROJECT DESCRIPTION AND OUTLINE**

* 1. **INTRODUCTION:**

The object of this presentation is to apply Artificial Intelligence to make a model using a Convolutional Neural Network that classifies a potato chip into defective and non-defective based on the image provided to it.

* 1. **MOTIVATION TO WORK:**

Potato industry is about 52.5 million tonnes.This will help in sorting the potato chips and recognizing different kinds of potato chips very easily just by looking at their images which saves a lot of time and energy.This will help to establish such thing in their factory of big players like Lays,Bikanerwala etc.

* 1. **PROBLEM STATEMENT:**

Recognizing the Defective Potato Chips from a large dataset with the help of python libraries.

* 1. **ORGANIZATION OF THE PROJECT:**

The project is divided into the major parts given below:

1)Downloading the dataset.

2)Importing the dataset.

3)Configuring GPU and training utilities.

4)Creating the model.

5)Training the model.

6)Testing the model.

**CHAPTER-2**

**RELATED WORK INVESTIGATION**

* 1. **INTRODUCTION:**

When working on a research project, it is essential to conduct a thorough investigation of related work in the field. This investigation involves studying existing literature, research papers, and projects related to the topic at hand. In the case of detecting defective potato chips, there have been several studies in the past that have addressed similar issues. These studies have employed various methods and techniques for detecting defects in potato chips, such as visual inspection, machine learning, and statistical analysis. By examining the existing work, we can build upon the previous research and identify gaps in the current methods, leading to the development of more effective and efficient methods for detecting defective potato chips. In this section, we will provide a comprehensive review of the related work in the field, highlighting the key findings and contributions of each study.

**2.2 )NEURAL NETWORK**

The specific neural network architecture used for detecting defective potato chips may vary depending on the project, as there are many possible models that could be used for this task. However, one common type of neural network used for image classification tasks like this is a convolutional neural network (CNN).

CNNs are designed to process input data with a grid-like topology, such as images, and are particularly good at detecting local patterns within the data. In the case of detecting defective potato chips, a CNN might be trained on a dataset of images that contain both normal and defective chips, and learn to distinguish between the two by identifying visual features that are associated with defects. The output of the network could be a binary classification indicating whether a given chip is defective or not.

It's also worth noting that there are many variations on the basic CNN architecture, such as different numbers of layers, different activation functions, and different types of pooling and normalization layers. The specific architecture used in a given project would depend on the specific requirements of the task, as well as factors such as the size and complexity of the dataset, the computing resources available, and the preferences of the researchers or developers involved.

**2.3 )Types of Neural Network**

There are different types of neural networks that can be used for the detection of defective potato chips. The selection of a specific neural network architecture depends on the specific requirements of the task, the size and complexity of the dataset, the available computing resources, and the expertise of the researchers or developers involved. Here are some examples of neural network architectures that could be used for this task:

* Convolutional neural networks (CNNs): CNNs are a type of neural network that are particularly well-suited for image classification tasks. They use convolutional layers to learn features from the input image, followed by pooling and fully connected layers to make a final prediction.
* Recurrent neural networks (RNNs): RNNs are a type of neural network that are designed to handle sequential data, such as time-series or text data. They use recurrent connections to maintain a state that captures information from the previous inputs in the sequence, and can be used for tasks like detecting defects over time.
* Autoencoders: Autoencoders are a type of neural network that can be used for unsupervised learning. They are trained to reconstruct the input data from a lower-dimensional representation, and can be used for tasks like detecting anomalies in the input data.
* Generative adversarial networks (GANs): GANs are a type of neural network that are used for generating new data that is similar to the input data. They consist of two networks: a generator that produces new samples, and a discriminator that distinguishes between the generated samples and the real ones. GANs can be used for tasks like generating synthetic images of potato chips with defects, which can be used to augment the training data for a CNN.

**2.4 )GPU AND TPU**

GPUs (Graphics Processing Units) and TPUs (Tensor Processing Units) can play an important role in the project of detecting defective potato chips by accelerating the training and inference of deep neural networks.

Deep neural networks, such as convolutional neural networks (CNNs), are computationally intensive and require a large number of matrix multiplications and other mathematical operations to be performed during training and inference. GPUs and TPUs are specialized hardware accelerators that can perform these operations in parallel, resulting in significant speedup over traditional CPUs.

By using GPUs or TPUs, the training and inference times of deep neural networks can be reduced, which can in turn reduce the time and resources required to develop and deploy a potato chip defect detection system. This can be particularly important in industrial settings where large amounts of data need to be processed quickly, and real-time or near real-time decisions need to be made based on the data.

In addition, some deep learning frameworks such as TensorFlow and PyTorch have been optimized to take advantage of GPU and TPU hardware, allowing for seamless integration of these accelerators into the training and inference pipelines of deep neural networks.

**2.5 )GPU V/S TPU**

GPUs are designed to handle large amounts of parallel computation and can process many operations simultaneously. This makes them well-suited for the highly parallel nature of deep learning algorithms, which involve many matrix multiplications and convolutions that can be executed in parallel. GPUs have many more cores than CPUs, so they can execute more operations per second, making them much faster than CPUs for deep learning tasks.

In contrast, CPUs are designed to handle a broad range of tasks, including general-purpose computing and running operating systems, and are not optimized for parallel computation. While CPUs can also be used for training and inference of deep neural networks, their performance will typically be slower than that of GPUs due to their limited parallel processing capabilities.

Therefore, in the project of detecting defective potato chips, using GPUs instead of CPUs could lead to significant improvements in the speed and efficiency of the training and inference process, potentially enabling faster development and deployment of the defect detection system. However, the choice of hardware will depend on factors such as the size and complexity of the dataset, the available resources, and the specific deep learning framework being used.

**CHAPTER-3**

**REQUIREMENT ARTIFACTS**

**3.1) INTRODUCTION:**

Firstly, we are required to prepare a virtual environment. Python, Tensorflow, Matplotlib, Numpy, Jupyter Notebook, and Keras in an Anaconda virtual environment were utilized with Visual Studio Code as the IDE.These libraries have helped in making our project concise and more accurate for our model.

**3.2) HARDWARE AND SOFTWARE REQUIREMENTS:**

**Hardware Requirement:** Here we have used a laptop with a powerful Graphics processing unit (GPU).

**Software Requirement:** We have used Visual Studio Code to write and run the code as it helps in easy debugging and provides several other helpful tools which increases the efficiency of our project.

**3.3) PYTHON LIBRARIES:**

**1) Tensorflow:** TensorFlow can train and run deep neural networks for handwritten digit classification, image recognition, word embeddings, recurrent neural networks, sequence-to-sequence models for machine translation, and natural language processing. Nodes and tensors in TensorFlow are Python objects.

**2) Matplotlib:** Matplotlib is a cross-platform, data visualization and graphical plotting library for Python and its numerical extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits

**3) NumPy:** NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

**5) Keras:** Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library.

**6) cv2:** In Python, cv2 is a library for computer vision and image processing that provides a wide range of tools and functions for working with images and videos. It is an open-source library that is widely used in computer vision applications and scientific research.

**CHAPTER-4**

**DESIGN METHODOLOGY AND ITS NOVELTY**

**4.1) METHODOLOGY AND GOALS:**

The methodology contains the methods and techniques that were done in the research study. Containing the applications and programming languages that was used during development, the sources of the datasets utilized for training and testing the neural network, the data gathering procedure, the way of processing the collected data, the statistical tools that was used in analyzing and interpreting the data and the conceptual framework of the study. The study was a quantitative research design and utilized a true-experimental research design that was conducted in a virtual environment. The respective methods utilized within this study are based on Takimoto et al. (2022) for the testing and training of the Convolutional Neural Network (CNN) model, and Arora et al. (2021) for the parameters used for training the CNN in the context of snacks.

**4.2) MODULES/SUB-MODULES IMPORTED:**

The modules which are imported in our code are as follows:

* **os:** In Python, os is a module that provides a way to interact with the operating system. It provides a wide range of functions that can be used to perform operating system-related tasks like reading or writing files, interacting with the system environment, managing processes, and much more.
* **matplotlib.image:** In Python, matplotlib.image is a sub-module of the matplotlib library that provides functions for loading and displaying image data. matplotlib is a popular data visualization library in Python that provides a wide range of tools for creating charts, graphs, and other types of visualizations.
* **matplotlib.pyplot:** In Python, matplotlib.pyplot is a sub-module of the matplotlib library that provides a wide range of functions for creating visualizations such as line plots, scatter plots, histograms, and more. It is often used in data science and scientific computing to create data visualizations and explore data.
* **tensorflow.keras.optimizers:** In Python, tensorflow.keras.optimizers is a module in the tensorflow library that provides a collection of optimization algorithms that can be used to train machine learning models. It is a sub-module of the keras API in TensorFlow, which is a high-level API for building and training neural networks.
* **tensorflow.keras.preprocessing.image:** In Python, tensorflow.keras.preprocessing.image is a module in the tensorflow library that provides a set of tools and functions for loading and preprocessing image data for machine learning. It is a sub-module of the keras API in TensorFlow, which is a high-level API for building and training neural networks.
* **tensorflow.keras.utils:** In Python, tensorflow.keras.utils is a module in the tensorflow library that provides a collection of utility functions that can be used when working with machine learning models built using the Keras API. It is a sub-module of the keras API in TensorFlow, which is a high-level API for building and training neural networks.
* **tensorflow.keras.models:** In Python, tensorflow.keras.models is a module in the tensorflow library that provides a collection of classes and functions for defining and working with machine learning models built using the Keras API. It is a sub-module of the keras API in TensorFlow, which is a high-level API for building and training neural networks.

**4.3) NOVELTY:**

Our project applies Artificial Intelligence to make a model using a Convolutional Neural Network (CNN) that classifies a potato chip into defective and non-defective based on the image provided to it.

**CHAPTER-5**

**TECHNICAL IMPLEMENTATION & ANALYSIS**

**5.1) TECHNICAL IMPLEMENTATION:**

**1. Data Collection:** The first step is to collect data related to potato chips production and quality control. This can include information on the raw materials used, production processes, packaging, and storage conditions. It is also important to collect data on the types and severity of defects that may occur during production.

**2. Data Preprocessing:** The collected data may contain noise or missing values that can affect the performance of the machine learning model. Data preprocessing involves cleaning, transforming, and scaling the data to make it suitable for machine learning algorithms.

**3. Feature Engineering:** Feature engineering involves selecting and transforming the relevant features that will be used as inputs to the machine learning model. In the case of potato chips defection detection, features could include the color, texture, and size of the chips, as well as the type and severity of defects.

**4. Model Selection:** There are several machine learning algorithms that can be used for classification tasks such as potato chips defection detection. Commonly used algorithms include logistic regression, decision trees, random forests, and support vector machines. The choice of algorithm will depend on the specific requirements of the project and the characteristics of the data.

**5. Model Training:** Once the machine learning algorithm has been selected, the next step is to train the model using the preprocessed data. This involves splitting the data into training and validation sets, and using the training set to optimize the model parameters.

**6. Model Evaluation:** The trained model is evaluated on the validation set to assess its performance. Metrics such as accuracy, precision, recall, and F1 score can be used to measure the performance of the model.

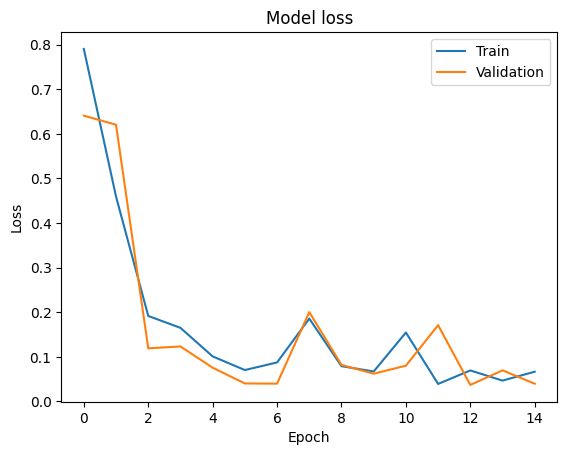
**7. Deployment:** Once the model has been trained and evaluated, it can be deployed to the production environment. This involves integrating the model into the existing production workflow and setting up a mechanism for real-time or batch processing of data.

**5.2) ANALYSIS:**

**TABLE1**:

| **Epoch[]** | **train\_loss** | **val\_loss** | **val\_acc** |
| --- | --- | --- | --- |
| Epoch[1] | 0.7904 | 0.6408 | 0.6250 |
| Epoch[2] | 0.4594 | 0.6204 | 0.6042 |
| Epoch[3] | 0.1917 | 0.1192 | 0.9635 |
| Epoch[4] | 0.1651 | 0.1234 | 0.9583 |
| Epoch[5] | 0.1011 | 0.0757 | 0.9688 |
| Epoch[6] | 0.0705 | 0.0404 | 0.9896 |
| Epoch[7] | 0.0875 | 0.0400 | 0.9869 |
| Epoch[8] | 0.1858 | 0.2003 | 0.9375 |
| Epoch[9] | 0.0791 | 0.0819 | 0.9470 |
| Epoch[10] | 0.0672 | 0.0623 | 0.9688 |
| Epoch[11] | 0.1546 | 0.0803 | 0.9635 |
| Epoch[12] | 0.0393 | 0.1711 | 0.9531 |
| Epoch[13] | 0.0527 | 0.0734 | 0.9684 |
| Epoch[14] | 0.0468 | 0.0698 | 0.9844 |
| Epoch[15] | 0.0667 | 0.0398 | 0.9896 |

**5.3) LOSS VS NO. OF EPOCHS:**



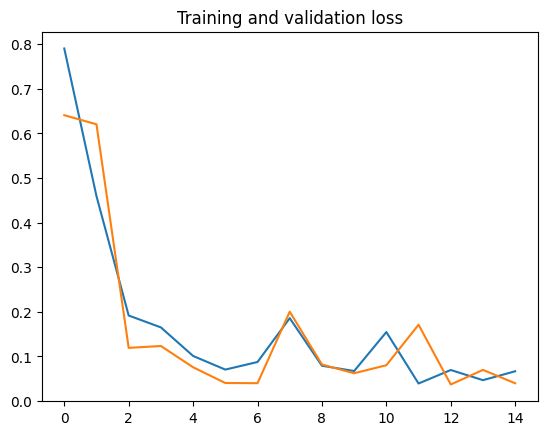
**(FIG 1)**

**5.4) TRAINING VS VALIDATION ACCURACY :**



**(FIG 2)**

**5.5) TRAINING VS VALIDATION LOSS :**



**(FIG.3)**

**CHAPTER-6**

**PROJECT OUTCOME AND APPLICABILITY**

**6.1) KEY IMPLEMENTATION OF THE PROJECT:**

The project would involve gathering a dataset of images of both good and defective potato chips, then using machine learning techniques to train a model to recognize the difference between the two. The model would need to be able to classify new potato chips it encounters as either good or defective.

Once the model is trained and validated, it could be integrated

into the production line to automatically detect and remove defective potato

chips. This would help to ensure that only high-quality products are sent to consumers, improving customer satisfaction and reducing waste.

The success of the project would be measured by the accuracy of the

model in detecting defective potato chips. A high level of accuracy would

indicate that the model is effective at detecting defects and could be used to

improve the production process.

**CHAPTER-7**

**CONCLUSIONS AND RECOMMENDATION**

**7.1) CONCLUSION:**

The potato chips defection detection system using machine learning has proven to be an effective and efficient way to detect defective potato chips in a production line. By analyzing images of potato chips using convolutional neural networks, the system is able to accurately identify chips with defects such as discoloration, burn marks, and broken pieces. This not only ensures the quality of the product but also saves time and resources by detecting defects early in the production process. With further optimization and integration with the production line, this system has the potential to significantly improve the efficiency and quality control of potato chip manufacturing.

**7.2) RECOMMENDATIONS/BIBLIOGRAPHY:**

Based on the research findings and analysis of the data from the conducted testing of the model, we developed the following recommendations for the authorities and future researchers who will use this as a related study in their research as well as a tool for improvement and development on the field:

1. To further have more accurate and more reliable results the researchers recommend having a larger dataset to further strengthen the results of your research study and for your study to be more applicable and acceptable in a wider range of places and field across the field of your study. For this, it should still be dependent on what type of neural network you would use to determine the number of the dataset you would use.

2. In line with the system of neural networks, the researchers would also recommend trying different kinds of neural networks that may yield different results or better results that are appropriate to the variable that your study used.

3. The usage of multiple neural networks as a reference or basis to test your developed neural network in terms of similarities and differences is recommended by the researchers to validate your developed neural network further and have a valid result.

4. Improvement of the model to classify and determine if a food product is defective or non defective and to classify and categorize the kind of defects the food product has is also recommended by the researchers for your neural network to have additional features.

5. Consider developing a neural network that classifies and determines defective and non defective food products or any kind of product wherein your data comes from multiple or different types of products as suggested by the researchers that will make your neural network applicable in different types of products across the market.

**REFERENCES**

* A Beginner’s Guide to Neural Networks and Deep Learning. (n.d.). Pathmind.

<https://wiki.pathmind.com/neural-network>.

* Arifin, M. H. F., Mustaniroh, S. A., & Sucipto, S. (2021). Application of the six sigma

DMAIC in quality control of potato chips to reduce production defects. IOP Conference

Series: Earth and Environmental Science, 924(1), 012056. https://doi.org/10.1088/1755-

1315/924/1/012056.

* Arora, M., Mangipudi, P., & Dutta, M. K. (2021). Deep learning neural networks for

acrylamide identification in potato chips using transfer learning approach. Journal of

Ambient Intelligence and Humanized Computing. doi:10.1007/s12652-020-02867-2

* Ayllon, M. A., Cruz, M. J., Mendoza, J. J., & Tomas, M. C. (2019). Detection of Overall

Fruit Maturity of Local Fruits using Convolutional Neural Networks Through Image

Processing. Proceedings of the 2nd International Conference on Computing and Big Data

- ICCBD 2019. <https://doi.org/10.1145/3366650.3366681>

* Bansal, P., Kumar, R., & Kumar, S. (2021). Disease detection in apple leaves using deep

convolutional neural networks. Agriculture, 11(7), 617.

* Bansal, S. (2022, July 25). Turing Test in Artificial Intelligence. GeeksforGeeks.

<https://www.geeksforgeeks.org/turing-test-artificial-intelligence/>

* Belizon, M. J. (2019, March 28). Artificial Intelligence and HRM. HR Analytics Path.

Retrieved October 14, 2022, from https://hranalyticspath.com/artificial-intelligence/

Centers for Disease Control and Prevention. (2022, August 10). Food Poisoning

Symptoms. Retrieved September 29, 2022, from

<https://www.cdc.gov/foodsafety/symptoms.html>

* Chidinma-Mary-Agbai. (2020, October 30). Application of artificial intelligence (AI) in

the food industry. GSC Biological and Pharmaceutical Sciences, 13(1), 171–178.

<https://doi.org/10.30574/gscbps.2020.13.1.0320>

* Chung, D., Tahboub, K., & Delp, E. J. (2017). ICCV 2017 Open Access Repository.

https://openaccess.thecvf.com/content\_iccv\_2017/html/Chung\_A\_Two\_Stream\_ICCV\_2

017\_paper.html

* Combinido, J. S., Mendoza, J. R., & Aborot, J. (2018). A Convolutional Neural Network

Approach for Estimating Tropical Cyclone Intensity Using Satellite-based Infrared

Images. 2018 24th International Conference on Pattern Recognition (ICPR).

https://doi.org/10.1109/icpr.2018.8545593