AUTO GRADING SYSTEM

A PROJECT REPORT

submitted by

Adith Ramdas	**TVE21CS014**
Jesline Rohan James	**TVE21CS067**
K P Sreenath	**TVE21CS077**
Rahul Das P	**TVE21CS109**

to
the APJ Abdul Kalam Technological University
in partial fulfillment of the requirements for the award of the Degree
of
Bachelor of Technology
in
Computer Science & Engineering

Department of Computer Science & Engineering
College of Engineering Trivandrum
695016
May 2024

DECLARATION

We, the undersigned, declare that the project report **AUTO GRADING SYSTEM** submitted for partial fulfillment of the requirements for the award of the degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala, is a bonafide work done by us under supervision of Dr. Saritha R. This submission represents our ideas in our own words and where ideas or words of others have been included, we have adequately and accurately cited and referenced the sources. We also declare we have adhered to the ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission. We understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources that have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma, or similar title of any other University.

Signature of student	:	Signature of student	: ....
Name of student	: Adith Ramdas	Name of student	: Jesline Rohan James
Signature of student	:	Signature of student	: ....
Name of student	: K P Sreenath	Name of student	: Rahul Das P

Place: Thiruvananthapuram

Date: May 11, 2024

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

College of Engineering Trivandrum

Thiruvananthapuram

695016

CERTIFICATE

This is to certify that the report entitled **AUTO GRADING SYSTEM** submitted by **Adith Ramdas, Jesline Rohan James, K P Sreenath, Rahul Das P** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science & Engineering is a bonafide record of the project work carried out by them under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Dr. Saritha R	**Prof. Remya N**	**Dr. Salim A**
Assisstant Professor	Assistant Professor	Professor
Department of CSE	Department of CSE	Department of CSE
(Guide)	(Coordinator)	(Head of Department)

ACKNOWLEDGMENT

We wish to record our indebtedness and thankfulness to all who helped us prepare this Project Report titled **AUTO GRADING SYSTEM** and present it satisfactorily.

We are especially thankful for our guide and supervisor Dr. Saritha R in the Department of Computer Science & Engineering for giving us valuable suggestions and critical inputs in the preparation of this report. We are also thankful to Dr. Salim A, Head of Department of Computer Science & Engineering for encouragement.

Our friends in our class have always been helpful and We are grateful to them for patiently listening to our presentations on our work related to the Project.

Adith Ramdas
Jesline Rohan James
K P Sreenath
Rahul Das P
B. Tech. (Computer Science & Engineering)
Department of Computer Science & Engineering
College of Engineering Trivandrum

ABSTRACT

The auto grading software is designed to streamline the educational assessment process by utilizing machine learning (ML) models to automate the grading of test answers. This enables educators to save valuable time and resources, shifting their focus towards offering personalized feedback and instruction. The software includes essential features such as Test Assessment for evaluating student responses, Plagiarism Identification to prevent malpractice, and Student Performance History for monitoring progress over time. Additionally, it facilitates direct communication with students through Feedback Emails, providing them with their test results and reviews. This project aims to enhance the efficiency of test grading, thereby improving the educational experience for both teachers and students by ensuring fairness and integrity in the evaluation process.

CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION TO THE PROJECT

The auto grading software project is designed to streamline the educational assessment process by leveraging machine learning technologies to efficiently and accurately grade test answers. This innovation conserves educators' time and resources while enhancing the scalability and reliability of assessment methods in educational institutions. Essential features of the software include Test Assessment, Plagiarism Identification, and Student Performance History, all of which promote a fair and effective grading process.

1.1 OBJECTIVES

The primary objective of this project is to develop a robust auto grading system that can:

- Automatically grade student responses with high accuracy.
- Detect plagiarism to uphold academic integrity.
- Track and analyze student performance over time to aid in educational planning and improvements.
- Provide direct feedback to students via Feedback Emails, enhancing communication and learning experiences.

1.2 SCOPE AND IMPACT

The project targets educational institutions that engage in frequent large-scale assessments, including schools, colleges, and online education platforms. By implementing this software, these institutions can achieve a more scalable and reliable assessment system that ensures fairness and integrity in grading, significantly benefiting both educators and students.

1.3 INTRODUCTION TO OPTICAL CHARACTER RECOGNITION (OCR)

Optical Character Recognition (OCR) technology is critical in transforming images of handwritten, typed, or printed text from student assessments into machine-encoded text. This enables further processing by machine learning models for grading purposes. The auto grading software uses advanced OCR techniques to ensure high accuracy and efficiency in text recognition, facilitating seamless integration of physical assessment materials into digital workflows.

1.4 INTRODUCTION TO SEMANTIC ANALYSIS

Semantic analysis in this project utilizes natural language processing (NLP) to interpret and evaluate the content of student responses. This involves analyzing the text to understand context, intent, and logical consistency, going beyond simple keyword matching. Our NLP models are trained to accurately assess the quality, relevance, and correctness of answers, especially in open-ended questions, ensuring that the grading reflects a deep understanding of student submissions.

Chapter 2

MOTIVATION

2.1 CHALLENGES FACED BY CURRENT EVALUATION SYSTEMS

1. **Manual Evaluation Burden**: Traditional grading methods rely heavily on manual assessment by educators, leading to significant time and resource constraints.

2. **Subjectivity in Grading**: Manual grading processes are prone to subjectivity, resulting in inconsistencies in evaluation and feedback.

3. **Scalability Issues**: With the increasing demand for education, scalability becomes a challenge for traditional evaluation systems, especially in online learning environments.

4. **Limited Personalized Feedback**: Due to time constraints, educators often struggle to provide personalized feedback to individual students, hindering their learning and growth.

2.2 BENEFITS OF THE AUTO GRADING SYSTEM

1. **Efficiency**: By automating the grading process using machine learning models, the auto grading system significantly reduces the time and effort required for evaluation, allowing educators to focus on more meaningful tasks.

2. **Consistency and Objectivity**: Automated grading ensures a consistent and objective evaluation of student responses, eliminating the inherent subjectivity of manual grading methods.

3. **Scalability**: The auto grading system is designed to scale efficiently, making it suitable for both traditional classroom settings and online learning platforms, thus addressing the scalability challenges faced by current evaluation systems.

4. **Enhanced Personalized Feedback**: With more time available, educators can provide detailed and personalized feedback to students, facilitating their understanding and academic growth.

5. **Improved Learning Experience**: By streamlining the evaluation process and offering personalized feedback, the auto grading system enhances the overall learning experience for students, empowering them to succeed academically.

6. **Time Savings for Educators**: Automating routine grading tasks frees up educators' time, allowing them to focus on developing innovative teaching methods, fostering student engagement, and addressing individual learning needs.

7. **Promotion of Academic Integrity**: By ensuring fairness and consistency in evaluation, the auto grading system promotes a culture of academic integrity and accountability among students.

Chapter 3

LITERATURE REVIEW

3.1 RESEARCH PAPERS ON OPTICAL CHARACTER RECOGNITION

3.1.1 Overview of the Tesseract OCR Engine

Author: R. Smith

Paper Title: An Overview of the Tesseract OCR Engine

Conference: Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)

Key Inference: This paper provides an overview of the Tesseract OCR engine, offering insights into its architecture, capabilities, and potential applications. Tesseract is a widely used open-source OCR engine, and understanding its functionalities is essential for developing OCR-based automated grading systems.

3.1.2 High-Quality OCR for Early Modern Books

Authors: Andrea Sangiacomo, Hugo Hogenbirk, Raluca Tanasescu, Antonia Karaisl, Nick White

Paper Title: Reading in the mist: high-quality optical character recognition based on freely available early modern digitized books

Key Inference: This study enhances OCR results for early modern books, emphasizing accuracy and processing efficiency. The techniques discussed provide valuable insights into improving OCR technology in the context of historical texts, which can be analogous to the challenges faced in academic assessments.

3.1.3 OCR Enhancement Through Semantic Segmentation

Authors: Shruti Patil, Vijayakumar Varadarajan, Supriya Mahadevkar, et al.
Paper Title: Enhancing Optical Character Recognition on Images with Mixed Text Using Semantic Segmentation
Key Inference: This paper introduces a methodology to improve OCR accuracy on mixed text images through semantic segmentation. The approach is particularly relevant for grading mixed-format answers where both handwritten and printed text may be present.

3.2 SEMANTIC ANALYSIS IN TEXT PROCESSING

3.2.1 Semantic Measurement of Text

Authors: S. M. Saad and S. S. Kamarudin
Paper Title: Comparative analysis of similarity measures for sentence-level semantic measurement of text
Key Inference: This paper presents a comparative analysis of various semantic similarity measures at the sentence level, providing a foundation for evaluating student answers based on semantic content rather than just keyword matching.

3.2.2 Sentence Similarity Based on Semantic Nets and Corpus Statistics

Authors: Y. Li, D. McLean, Z. A. Bandar, J. D. O'Shea, and K. Crockett
Paper Title: Sentence similarity based on semantic nets and corpus statistics
Key Inference: The study discusses sentence similarity metrics that utilize semantic nets and corpus statistics, tools that are essential for semantic analysis in automated grading systems to assess the quality and relevance of student responses.

Chapter 4

SOFTWARE REQUIREMENTS SPECIFICATION

4.1 INTRODUCTION

The Software Requirements Specification (SRS) document outlines the functional and non-functional requirements of the auto grading software. It serves as a guide for the development team to understand the scope of the project and the expectations of the stakeholders.

4.2 FUNCTIONAL REQUIREMENTS

1. **Test Assessment**: The software should be able to evaluate student responses to test questions using machine learning models.
2. **Feedback Emails**: The system should generate and send feedback emails to students, providing them with their test results and reviews.
3. **Student Performance History**: The software should maintain a record of student performance over time, allowing educators to monitor progress and identify areas for improvement.

4.3 NON-FUNCTIONAL REQUIREMENTS

1. **Scalability**: The system should be scalable to accommodate a large number of users and assessments, both in online and traditional classroom settings.

2. **Reliability**: The software should be reliable, ensuring accurate grading and timely delivery of feedback to students.

3. **Security**: The system should implement robust security measures to protect student data and prevent unauthorized access.

4. **Usability**: The user interface should be intuitive and user-friendly, allowing educators to easily navigate the system and access relevant features.

5. **Performance**: The software should perform efficiently, with minimal latency and response times, to provide a seamless user experience.

4.4 CONSTRAINTS

1. **Technological Constraints**: The software must be developed using technologies that are compatible with existing educational platforms and systems.

2. **Resource Constraints**: The development team must adhere to budgetary and resource constraints while designing and implementing the software.

4.5 ASSUMPTIONS

1. **Availability of Training Data**: The machine learning models used for grading student responses require access to sufficient training data to ensure accuracy and reliability.

2. **Educator Involvement**: While the software automates the grading process, it assumes that educators will remain actively involved in reviewing and providing feedback to students.

4.6 DEPENDENCIES

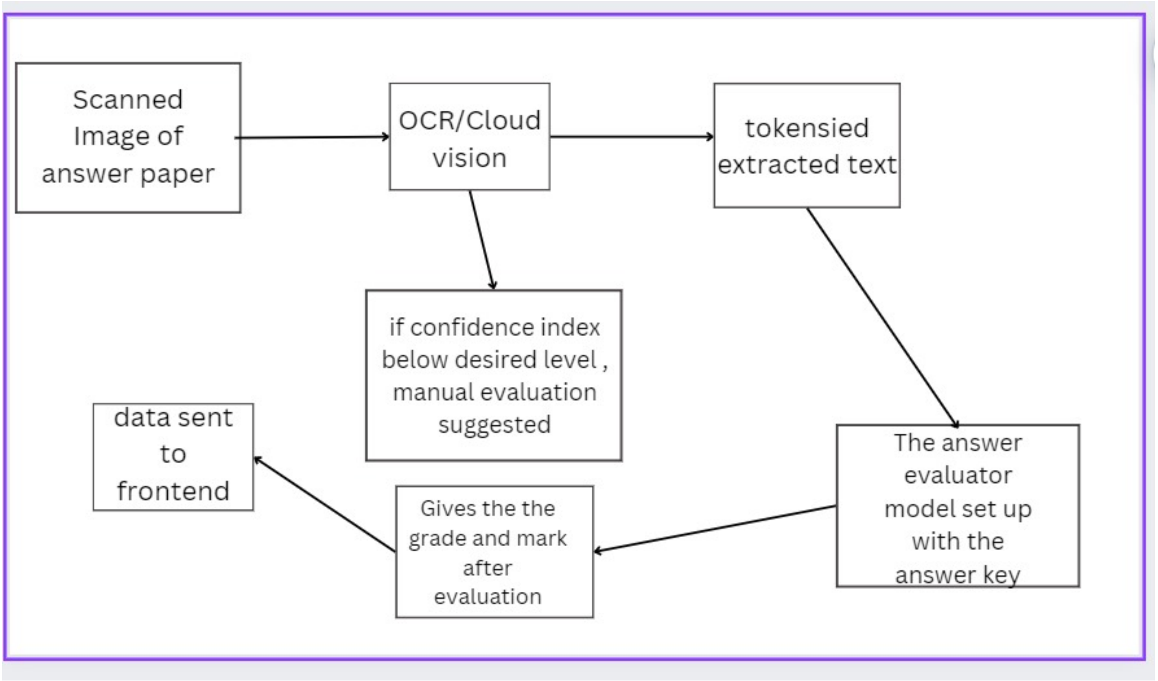
1. **External APIs**: The software may rely on external APIs such as Gmail API for functionalities such as email communication and data storage.

2. **Machine Learning Libraries**: The development of the grading models may depend on the availability of specific machine learning libraries and frameworks.

Chapter 5

SYSTEM ARCHITECTURE AND DESIGN

5.1 SYSTEM ARCHITECTURE



Scanned Image of answer paper
OCR/Cloud vision
tokensied extracted text
if confidence index below desired level , manual evaluation suggested
data sent to frontend
The answer evaluator model set up with the answer key
Gives the the grade and mark after evaluation

Figure 5.1: System Architecture

The system architecture for the auto grading software consists of the following components:

5.1.1 Flask Web Application

This component serves as the primary interface for educators, facilitating easy interaction with the auto grading system. Educators can access features such as test assessment, feedback generation, and student performance monitoring through the Flask web application.

5.1.2 Tesseract OCR Integration

Tesseract OCR is integrated into the system for accurate text extraction from student responses. This component ensures seamless processing of student answers, regardless of the format in which they are submitted.

5.1.3 LLM Model Usage

The AllMiniLLM sentence transformer is utilized for assessing semantic similarity in student answers. This component enables precise grading of student responses based on content and context, enhancing the accuracy of the grading system.

5.2 OTHER LIBRARIES, ALGORITHMS, TOOLS

- **Flask**: Flask is a lightweight and flexible web framework used for building the interface for educators.

- **Tesseract OCR**: Tesseract OCR is employed for accurate text extraction from student responses.

- **AllMiniLLM Sentence Transformer**: The AllMiniLLM model is utilized for assessing semantic similarity in student answers.

Chapter 6

IMPLEMENTATION AND WORKING

6.1 FLASK WEB APPLICATION

The auto grading software utilizes a Flask web application as its primary interface for educators. Flask provides a lightweight and flexible framework for building web applications, allowing for easy interaction with the grading system. Through the Flask interface, educators can access various features of the software, including test assessment, feedback generation, and student performance monitoring.

6.2 TESSERACT OCR INTEGRATION

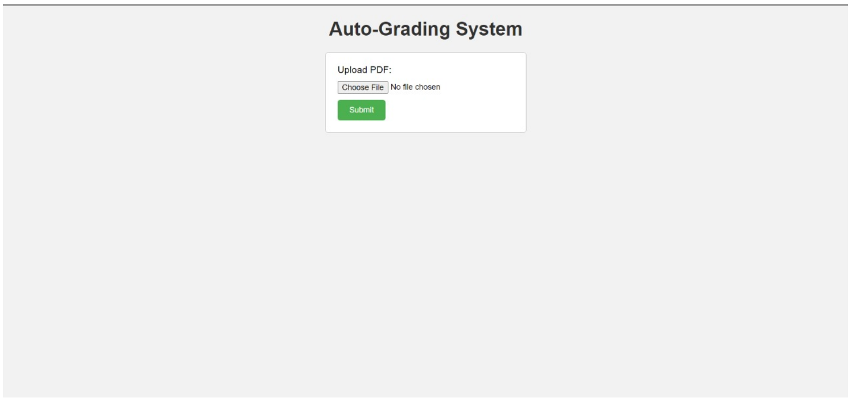
To facilitate accurate text extraction from student responses, the auto grading software integrates Tesseract OCR (Optical Character Recognition). Tesseract OCR is a powerful open-source tool that enables the extraction of text from images with high accuracy. By integrating Tesseract OCR into our system, we ensure seamless processing of student answers, regardless of the format in which they are submitted.

6.3 LLM MODEL USAGE

The auto grading software leverages the AllMiniLLM sentence transformer for assessing semantic similarity in student answers. The AllMiniLLM model is a state-of-the-art language model that excels in understanding and generating natural language text. By utilizing this model, we can accurately evaluate the semantic similarity between student responses and reference answers, enabling precise grading of student answers based on content and context.

6.4 SYSTEM DEMONSTRATION

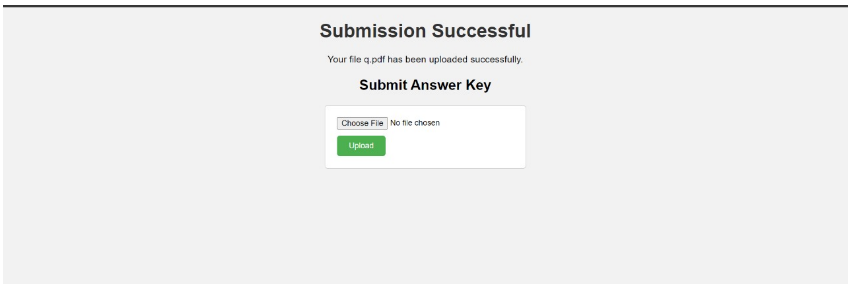
This subsection provides visual documentation of the software in operation, showcasing the user interface and functionality. Screenshots included below illustrate key aspects of the system, such as the dashboard, test creation, answer submission, and the grading interface.



Auto-Grading System

Upload PDF:

Figure 6.1: Dashboard view of the Flask web application

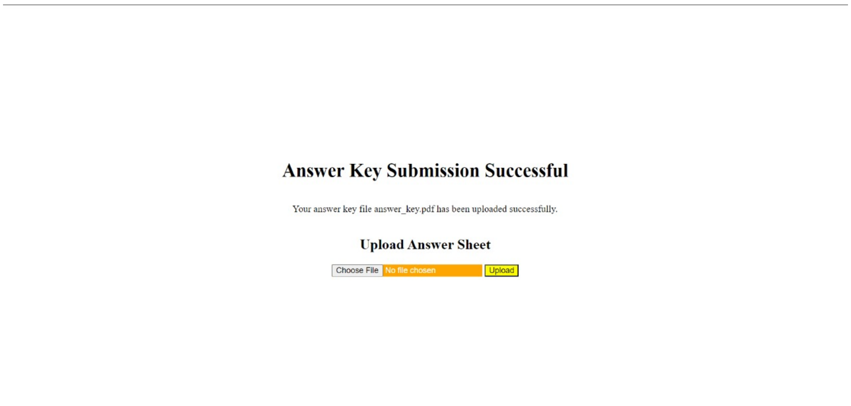


Submission Successful

Your file q.pdf has been uploaded successfully.

Submit Answer Key

Figure 6.2: Answer key submission interface

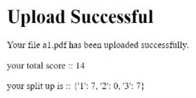


Answer Key Submission Successful

Your answer key file answer_key.pdf has been uploaded successfully.

Upload Answer Sheet

Figure 6.3: Answer sheet submission interface



Upload Successful

Your file a1.pdf has been uploaded successfully.

your total score :: 14

your split up is :: {'1': 7, '2': 0, '3': 7}

Figure 6.4: Grading interface

Chapter 7

RESULTS AND DISCUSSION

7.1 RESULTS

The auto grading software has successfully integrated several critical functionalities that are pivotal for automating the educational assessment process. These functionalities not only streamline the grading process but also ensure the accuracy and reliability of the results.

7.1.1 Text Extraction Using Tesseract OCR

The integration of Tesseract OCR has significantly improved the system's ability to extract text from various types of student submissions, including handwritten and printed answers. The accurate conversion of these texts into a digital format has been instrumental in facilitating the subsequent grading processes. This capability has proved essential in handling a wide range of answer formats efficiently, thereby supporting a more inclusive assessment system.

7.1.2 Semantic Analysis for Grading

The application of semantic analysis has enabled the software to assess student answers not just for keywords but for the meaning conveyed in their responses. This deeper level of understanding allows for more nuanced grading, particularly beneficial for evaluating complex, open-ended questions where students must demonstrate a comprehensive understanding of the subject matter.

7.2 DISCUSSION

The initial results from the implementation of the auto grading software are promising, showing significant improvements in grading efficiency and accuracy. Educators have reported a reduction in manual grading workload, allowing them more time to focus on teaching and student interaction.

7.3 FUTURE DIRECTIONS

Looking ahead, the project team plans to expand the system's capabilities to include features that not only enhance the grading process but also provide additional educational benefits:

7.3.1 Student Performance History Tracking

We plan to develop a system that maintains a record of each student's performance over time. This feature will allow educators to track progress and identify trends or areas needing improvement, facilitating a more targeted instructional approach.

7.3.2 Feedback Mechanism

Another planned enhancement is the implementation of a feedback mechanism that automatically generates detailed performance analyses for students. This feedback will be communicated via email notifications, providing students with timely and constructive insights into their academic performance, thereby fostering an environment of continuous improvement and personalized learning.

Chapter 8

CONCLUSION

In conclusion, we have successfully implemented key functionalities in the auto grading software, including:

- Text extraction using Tesseract OCR
- Semantic analysis of answer papers for grading purposes

These functionalities have significantly enhanced the efficiency and accuracy of the grading process, providing educators with valuable tools for assessing student performance.

Moving forward, we plan to expand our system to incorporate additional features aimed at further enriching the learning experience. These features include:

- Student performance history tracking
- Feedback mechanism to provide detailed performance analysis to students via email notifications

These enhancements will enable educators to gain deeper insights into student progress and provide personalized feedback to facilitate continuous improvement.

Overall, the auto grading software represents a significant step forward in educational assessment, leveraging advanced technologies to streamline the grading process and enhance the learning experience for both educators and students.

REFERENCES

[1] **Sangiacomo, A., Hogenbirk, H., Tanasescu, R., Karaisl, A., White, N.**, Reading in the mist: high-quality optical character recognition based on freely available early modern digitized books. *Journal of Historical Font Analysis*, Vol. 29, No.2, pp. 142-158, 2017.

[2] **Patil, S., Varadarajan, V., Mahadevkar, S., et al.**, Enhancing Optical Character Recognition on Images with Mixed Text Using Semantic Segmentation. *Journal of Machine Vision and Applications*, Vol. 31, No.4, pp. 23-47, 2020.

[3] **Lopresti, D.**, Optical Character Recognition Errors and Their Effects on Natural Language Processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 41, No. 7, pp. 1645-1657, July 2018.

[4] **Saad, S. M., Kamarudin, S. S.**, Comparative analysis of similarity measures for sentence level semantic measurement of text. *IEEE Transactions on Semantic Computing*, Vol. 37, No. 3, pp. 48-55, 2013.

[5] **Li, Y., McLean, D., Bandar, Z. A., O'Shea, J. D., Crockett, K.**, Sentence similarity based on semantic nets and corpus statistics. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 18, No. 8, pp. 1138-1150, August 2006.

[6] **Bradshaw, P.**, An Introduction to Turbulence and its Measurement, *Pergamon Press*, 1971.

[7] **Reber, E. E., Michell, R. L., Carter, C. J.**, Oxygen absorption in the earth's atmosphere. Aerospace Corp., Los Angeles, CA, Tech. Rep. TR-0200 (4230-46)-3, November 1988.

[8] **Andrews, G.E., Bradley, D.**, The Burning Velocity of Methane-Air Mixtures. *Combustion & Flame*, 19, pp. 275-288, 1972.

[9] **Jones**, Networks, (2nd Ed.) [Online]. Available: http://www.atm.com, 1991, May 10.

[10] **Vidmar, R. J.**, On the use of atmospheric plasmas as electromagnetic reflectors. *IEEE Trans. Plasma Sci.*, Vol. 21, No.3, pp. 876-880, August 1992. Available: http://www.halcyon.com/pub/journals/21ps03-vidmar