Aktualisierbare Braillezeile mit Raspberry Pi und Arduino

bei mir unter: *Braillezeile mit Servos-Artikel* unter Servos als pdf gespeichert.

International Journal of Current Engineering and Technology E-ISSN 2277 – 4106, P-ISSN 2347 – 5161

©2016 INPRESSCO®, All Rights Reserved Available at http://inpressco.com/category/ijcet

Research Article

965| International Journal of Current Engineering and Technology, Vol.6, No.3 (June 2016)

Refreshable Braille Display using Raspberry Pi and Arduino

Saurabh Bisht†\*, Sandeep Reddy Goluguri† , Rajat Maheshwari†, Akhilesh Kumar†and P.Sathya†

†School of Electronics Engineering, VIT University, Vellore, Tamilnadu-632014, India

Accepted 01 June 2016, Available online 06 June 2016, Vol.6, No.3 (June 2016)

Abstrakt

Das Ziel dieses Projekts ist es, eine aktualisierbare elektronische Braillezeile mit Raspberry Pi 2 und Arduino Mega2560 Board zu erstellen. Dies zielt darauf ab, eine aktualisierbare Braille-Anzeige zu erstellen, die in der Lage ist, normale Textdateien sowie gedruckte Bilddateien in Braille umzuwandeln. Um die Funktionalität zum Konvertieren von Bilddateien in Text zu erreichen, haben wir die tesseract-ocr-Engine von Google verwendet, die derzeit die beste verfügbare OCR-Engine ist und sehr genau ist. Der Raspberry Pi übernimmt die Bildverarbeitung und der Arduino steuert das Display. Das Display besteht aus 6 Servomotoren, die per PWM angesteuert werden können. Der Servomotor fungiert in diesem Projekt als Stellglied.

1. Einleitung

Braille ist eine Schreib- und Lesemethode, die von blinden oder sehbehinderten Menschen verwendet wird. In Braille-Schrift wird jedes Zeichen durch eine eindeutige Kombination von 6 Punkten dargestellt, die eine Zelle bilden. Es war schon immer eine Herausforderung, Braille-Bücher für Blinde zu machen. Im Durchschnitt dauert eine Seite mit normalem Text etwa 4 Seiten (A. Domale, B. Padalkar, R. Parekh und M. A. Joshi et al., 2013) in Blindenschrift, wodurch Braille-Bücher oft ziemlich sperrig werden, was höhere Produktionskosten mit sich bringt. Um dieses Problem zu umgehen, haben wir uns eine Lösung ausgedacht, um eine Braille-Anzeige zu erstellen, die sowohl Text- als auch Bildeingaben aufnehmen und in Braille umwandeln kann. Derzeit sind viele aktualisierbare Braillezeilen auf dem Markt erhältlich, aber alle stehen vor einem großen Problem, nämlich hohen Kosten. Daher schlagen wir ein Design vor, um diese Herausforderung anzugehen.

In der Vergangenheit gab es einen Artikel, der ein Design einer Braille-Anzeige unter Verwendung von Mikroprozessoren vorschlug, aber dieses Gerät zielte darauf ab, den Braille-Text in eine Anzeige umzuwandeln, die eine mechanische piezoelektrische Anzeige hat (S. M. Elster, B. L. Zuber und J. L. Trimble et al., 1985). Die größte Herausforderung dabei war, dass wir nicht viele Bücher/Lesematerialien in Blindenschrift haben. Auf der jährlichen IEEE-Konferenz INDICON 2012 wurde ein Gerät namens DRISHTI (Vineeth Kartha; Dheeraj S. Nair; S. Sreekant; P. Pranoy; P. Jayaprakash et al., 2012) vorgeschlagen, das über ein gestenbasiertes Braille-Gerät diskutierte, das Gerät war unter Verwendung von Solenoiden als Aktuatoren implementiert. Arduino IDE und Processing IDE wurden zusammen mit dem Atmega 328 Microcontroller verwendet. Zur Darstellung von Braille-Punkten wurde das Prinzip der Punktmatrix verwend Solenoide, obwohl das Design recht gut war, waren jedoch mit einigen Problemen wie hohem Stromverbrauch und Lärm konfrontiert, da Solenoide beteiligt waren. Kentaro Noda, Kiyoshi Matsumoto, Isao Shimoyama (Noda, Kentaro, Kiyoshi Matsumoto und Isao Shimoyama, et al., 2008) haben einen auf einer Hauttypstruktur basierenden Tastsensor vorgeschlagen und hergestellt, um die Objektoberfläche durch Scannen zu erfassen. Der vorgeschlagene Sensor hatte verschiedene Schichten von Elastomeren als Epidermis-Dermis, piezoresistive Ausleger wurden hinter dem Grat platziert, die für eine einzigartige Verformung des Sensors verantwortlich waren. Der vorgeschlagene Sensor war ziemlich effizient, erforderte jedoch viel Präzision, um den genauen Winkel zu scannen. Increasing Braille Literacy: Voice-Assisted Electronic Braille Books (Mahmoud Al-Qudsi et al,2013) (eBraille eBook) for the Visually Impaired Dieses Papier zielt darauf ab, eBooks über eine Braillezeile plus eine zusätzliche Sprachunterstützung, die ähnlich wie out ist, in Brailleschrift umzuwandeln Texteingabe, wobei direkter Text als Eingabe für die Datei angegeben werden kann. Hier schlagen wir also ein Design vor, das die oben genannten Herausforderungen löst. Die von uns entworfene Text-zu-Braille-Anzeige verwendet einen SBC (Raspberry Pi) und ein Arduino-Board, das jeden Text oder jedes Bild in Braille konvertieren kann.

Ein Himbeer-Pi ist ein SBC mit großer Rechenleistung. Raspberry Pi ist ein Einplatinencomputer in Kreditkartengröße, der von der Raspberry Pi Foundation entwickelt wurde und an einen Monitor oder Fernseher angeschlossen werden kann. Es wurde hauptsächlich entwickelt, um Grundschulkindern das Programmieren beizubringen, aber mit der Zeit hat sich pi seinen Ruf als Multitasking-Board aufgebaut, das von verschiedenen Bastlern und Herstellern für ihre Projekte verwendet wird. Das Board kann als vollwertiger Desktop fungieren und Videos in hoher Qualität verarbeiten. Es hat sogar die Fähigkeit, mit der Außenwelt zu interagieren und die Musiksysteme, den Fernseher usw. zu steuern. Hier verwenden wir

Raspberry Pi 2 Model B mit 1 GB RAM und 32-Bit-ARMv7-Prozessor.



Fig.1 Raspberry pi

Arduino ist ein Open-Source-Motherboard, das interaktive Hardwareprojekte entwerfen soll, die die Außenwelt durch Sensoren und Aktoren erfassen und steuern können. Arduino ist ein Motherboard, das auf einem 8-Bit-Atmega-Mikrocontroller arbeitet. Für unser Projekt verwenden wir Arduino mega2560 ist ein Mikrocontroller-Board auf ATmega2560.

Das Board hat viele digitale Pins und einige analoge Pins, die einen eingebauten Analog-Digital-Wandler auf dem Board haben. Das Board wird im Allgemeinen über die Arduino IDE programmiert, basierend auf dem Verarbeitungsprojekt, das C-, C++- und Java-Sprachen für die Programmierung unterstützt. Das Board hat nur wenige KB RAM, wodurch es nicht für Aktivitäten mit hoher Rechenleistung geeignet ist. Das Board wird von einem 16-MHz-Takt angetrieben, jedoch ohne Betriebssystem. Es kann nur über serielle Kommunikation programmiert werden, entweder über UART oder USB-Kabel. Es unterstützt eine Stromaufnahme von 5 V bis 20 V, wobei für unser Projekt 9 V optimal sind. Das in diesem Projekt verwendete Arduino Mega-Board verfügt über 54 digitale Ein- / Ausgangspins, von denen 15 Pins als PWM-Ausgänge verwendet werden können.



Fig.2 Arduino Mega

Vorgeschlagenes aktualisierbares Braille-System

Hardware-Stufe

Wir werden die folgenden Komponenten verwenden

1) Himbeer-Pi

2) Arduino-Board

3) Servomotoren

Flussdiagramm:

Flussdiagramm

Wir haben den Pi für dieses Projekt im Headless-Modus betrieben, wobei der Pi nicht mit einem Monitor verbunden ist. Der hier verwendete Raspberry Pi 2 verfügt über 4 USB-Anschlüsse, von denen einer für die serielle Kommunikation mit Arduino verwendet wird. Dieser Minicomputer arbeitet mit einem gewaltigen Maschinentakt von 900 MHz, der bis auf 1 GHz übertaktet werden kann und verfügt über einen Quad-Core-Prozessor. Das Gerät arbeitet mit einer 5-V-Stromversorgung. Wir haben an Raspbian OS gearbeitet, das auf Debian Linux basiert. Dieses Betriebssystem wurde speziell für Raspberry Pi entwickelt und wird auf diesem Board am häufigsten verwendet. Dieses Betriebssystem wird mit vorinstalliertem Python geliefert, das die Bildverarbeitung und die Arduino-Steuerung übernimmt. Das gesamte System läuft auf der Micro-SD-Karte, auf der sich das Betriebssystem und die Dateien befinden.

Es wird zum Verarbeiten des Bildes und zum Aufteilen des Bildes in Zeichen unter Verwendung von OCR-Techniken (Optical Character Recognition) verwendet. OCR ist ein System, das es uns ermöglicht, die handschriftlichen Notizen oder gedruckten Materialien in eine computerbearbeitbare Datei umzuwandeln, die in Zukunft verwendet werden kann. Mit diesen OCR-Techniken wandelt der pi also das Bild um und dann wird jedes Zeichen zugewiesen durch einen eindeutigen Code. Der Code wird an Arduino gesendet, das die Anzeige steuert.

Arduino wird seriell vom Himbeer-Pi unter Verwendung der Nanpy-Bibliotheken gesteuert. Die Nanpy-Bibliotheken sind in Python entworfen, die verwendet werden können, um Arduino über die serielle Schnittstelle zu steuern. Die Nanpy-Bibliotheken steuern den Arduino mit einer Baudrate von 115200 bps und mit der auf Arduino installierten Nanpy-Firmware. In diesem Prozess fungiert der Pi also als Master und steuert Arduino als Slave. Nachdem das Bild verarbeitet und die Zeichen in einen eindeutigen Code umgewandelt wurden, erhält es die Informationen über die Pins des Servomotors und benachrichtigt das Arduino über die Funktionen, die es zu tun hat. Dann erzeugt der Arduino PWM-Impulse, um die Aktuatoren zu steuern.

Zum Anzeigen der Ausgabe verwenden wir eine Kombination von sechs Servomotoren, da alle Braille-Kombinationen aus verschiedenen Kombinationen von sechs Punkten gebildet werden.

Der Servomotor unterscheidet sich von einem normalen Gleichstrommotor, da er sich nur um 180 Grad drehen kann. Der Servomotor wird häufig in Robotikprojekten eingesetzt. Die Servomotoren benötigen eine 5-V-DC-Versorgung und ein Steuersignal, das auf Pulsweitenmodulation basiert, um die Position zu steuern. Pulsweitenmodulation ist eine Technik, bei der die Einschaltzeit eines Signals in Bezug auf die Gesamtsignalzeit gesteuert wird. Hier hat das verwendete Signal eine Gesamtzeit (Zeit an plus Zeit aus) von 20ms. Die Zeit, für die der hohe Impuls gegeben wird, bestimmt also die Position, zu der der Servomotor gehen soll.

 Für eine Dauer von 0,5 ms eines 5-V-Impulses muss der Servomotor in die 0-Grad-Position gehen.

 Für eine Dauer von 1,5 ms eines 5-V-Impulses muss der Servomotor in die 90-Grad-Position gehen.

 Für eine Dauer von 2,5 ms eines 5-V-Impulses muss der Servomotor in die 180-Grad-Position gehen.

Nach Erhalt des Eingangsimpulses erhält der Servomotor die aktuelle Position des Motors und entscheidet somit über die Richtung und den Zeitraum, für den sich das angebrachte Rad drehen muss, wodurch das Rad oder der Zeiger (in diesem Fall) an die gewünschte Position gesendet wird.

Wir haben diese sechs Motoren in einer Box mit einem Schlitz ausgerichtet gehalten, so dass im EIN-Zustand dieser Motoren die Spitze des Zeigers den Finger einer blinden Person berührt und er anhand des durch die Spitzen gebildeten Musters erkennen kann, welches Zeichen wird angezeigt.

Software-Phase

Für die Codierung haben wir die Programmiersprache Python verwendet, die eine einfache, leicht verständliche Programmiersprache auf hohem Niveau ist. Das Python-Skript nutzt die Tesseract-OCR-Bibliotheken von Google zur Verarbeitung der Bilddatei. Tesseract-OCR-Bibliotheken sind eine der besten verfügbaren Bibliotheken zum Lesen des gedruckten Textes mit der geringsten Fehlerquote. Die OCR-Bibliotheken erzeugen also die Ausgabe als Textdatei, die die Ausgabe des Bildes in Form von einfachem Text ist (Patel, Chirag, Atul Patel und Dharmendra Patel et al, 2012).

Der aus den Tesseract-OCR-Bibliotheken gewonnene Text wird dann in einzelne Zeichen zerlegt und anschließend mit einem eindeutigen Code aus Nullen und Einsen versehen, der die Position von Servomotoren anzeigt. Dann wird der eindeutige Code seriell an den Arduino gesendet, der den Servomotor über den PWM-Ausgang steuert. Zu diesem Zweck verwenden wir die nanpy-Bibliotheken. Die Nanpy-Bibliothek verwendet Python, um die Steuerung der an den Arduino gesendeten seriellen Daten und die Aktionen, die der Arduino ausführt, zu steuern. Zu diesem Zweck hat Arduino Nanpy-Firmware installiert, die dem Raspberry Pi die vollständige Kontrolle über Arduino geben würde. So sendet das Python-Skript nach der Konvertierung des Bildes in Zeichen die Steuersignale an die entsprechenden Servos, die mit dem Arduino-Board verbunden sind, um die Servomotoren in die erforderliche Position zu bringen.

Neben Nanpy und Tesseract-ocr verwenden wir auch Pyserial, das hilft, Daten seriell an das Arduino-Board zu übertragen.

Arbeiten

Es gibt 2 Arbeitsmodi

1) Texteingabebasiert

2) Bildeingangsbasiert

Bei der auf Texteingabe basierenden Methode geben wir den gewünschten anzuzeigenden Text in eine Textdatei ein, die vom Python-Skript gelesen wird, und dann wird der Text in Zeichen zerlegt und seriell zur Anzeige an das Arduino gesendet.

Bei der bildbasierten Eingabe geben wir das gewünschte Bild als Eingabe ein und Tesseract-Ocr konvertiert das Bild in eine Textdatei mit dem erforderlichen Text darin.

 Die Textdatei wird als Eingabe für den Python-Code angegeben.

 Das Skript liest die Textdatei Zeichen für Zeichen.

 Für jedes einzigartige Zeichen generieren wir einen einzigartigen Code.

 Der eindeutige Code wird in ein Steuersignal umgewandelt.

 Diese Steuersignale werden seriell an den Arduino übertragen.

 Gemäß den Steuersignalen ändern die Servomotoren ihre Position und berühren die Finger der Person

Wir können auch die Betriebsgeschwindigkeit der Servomotoren gemäß den Bedürfnissen verschiedener Personen ändern, indem wir die Zeitverzögerung zwischen den aufeinanderfolgenden Zeichen erhöhen oder verringern

Bild

Fazit

Dieser Bild-zu-Braille-Konverter kann mit einem Kameramodul hinzugefügt werden und somit können die Bilder in einem regelmäßigen Zeitintervall erfasst und dann die Bilder als Text auf der aktualisierbaren Anzeige dargestellt werden, was es blinden Menschen ermöglicht, normale Bücher und Zeitschriften bequem zu lesen. Außerdem müssen sie nicht warten, bis die Braille-Version des Buches oder der Zeitschrift herauskommt. Unser Modell des aktualisierbaren Displays hat eine Geschwindigkeit zur Darstellung der Braille-Zeichen mit einer enormen Geschwindigkeit von 12 Zeichen in 2,5 Sekunden und durchschnittlich 290 Zeichen in 60 Sekunden.

Unser komplettes System arbeitet mit einer Stromversorgung von nur 15,4 Watt. Der Himbeer-Pi wird hier von einem 5-V-, 2-A-Ladegerät mit Strom versorgt, in dem pi die Verarbeitung durchführt und 2 Servomotoren über pi mit Strom versorgt werden. Der Arduino ist mit einem 9-V-0,6-A-Ladegerät verbunden, das Arduino dabei unterstützt, PWM-Signale zu erzeugen, und 4 Servomotoren werden von der Arduino-Platine mit Strom versorgt.





Verweise

A.Domale, B.Padalkar, R.Parekh und MAJoshi (2013), „Printed Book to Audio Book Converter for Visually Impaired“, India Educators Conference (TIIEC) Texas Instruments, Bangalore doi:10.1109/TIEC.2013.27, S. 114-120

S. M. Elster, B. L. Zuber und J. L. Trimble (1985), "A Research Prototype of an Inkbraille Reading Aid", in IEEE Transactions on Biomedical Engineering, vol. BME-32, Nr. 8, S. 633-636

Vineeth Kartha; Dheeraj S. Nair; S. Sreekant; P. Pranoy; P. Jayaprakash (2012), „DRISHTI – Ein gestengesteuerter Text-zu-Braille-Konverter“, India Conference (INDICON), 2012 Annual IEEE, S. 335-339

Noda, Kentaro, Kiyoshi Matsumoto und Isao Shimoyama (2008), „Taktiler Sensor mit stehenden piezoresistiven Auslegern, bedeckt mit zweischichtigen hautartigen Strukturen zur Texturerkennung der Objektoberfläche.“, Intelligent Robots and Systems, 2008. IROS 2008. IEEE/ Internationale RSJ-Konferenz „Aktualisierbare Braillezeile“ Patent US 20130203022 A1. 8. August 2013

Patel, Chirag, Atul Patel und Dharmendra Patel (2012) "Optische Zeichenerkennung durch das Open-Source-OCR-Tool Tesseract: Eine Fallstudie." Internationale Zeitschrift für Computeranwendungen 55.10 (2012)

https://en.wikipedia.org/wiki/Braille

http://www.pyroelectro.com/tutorials/servo\_motor/schematic.html