ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Automation of Object Sorting System Using Pick & Place Robotic Arm & Image Processing

Swarnendu Sarkar
M.Tech Student,
Department of ECE,
St.Martins Engineering College, JNTUH,Hyderabad.

K. Yadaiah
HOD & Associate Professor,
Department of ECE,
St.Martins Engineering College, JNTUH,Hyderabad.

Abstract:

The paper presents a smart approach for a real time inspection and selection of objects in continuous flow. Image processing in today's world grabs massive attentions as it leads to possibilities of broaden application in many fields of high technology. The real challenge is how to improve existing sorting system in the modular processing system which consists of four integrated stations of identification, processing, selection and sorting with a new image processing feature. Existing sorting method uses a set of inductive, capacitive and optical sensors do differentiate object color. This paper presents a mechatronics color sorting system solution with the application of image processing. Image processing procedure senses the objects in an image captured in real-time by a webcam and then identifies color and information out of it. This information is processed by image processing for pick-and-place mechanism.The Project deals with an automated material handling system. It aims in classifying the colored objects by colour, size,character which are coming on the conveyor by picking and placing the objects in its respective pre-programmed place. Thereby eliminating the monotonous work done by humans, achieving accuracy and speed in the work. The project involves sensors that senses the object's colour, size and sends the signal to the microcontroller. The microcontroller sends signal to circuit which drives the various motors of the robotic arm to grip the object and place it in the specified location. Based upon the detection, the robotic arm moves to the specified location, releases the object and comes back to the original position.

Keywords:
Camera, Conveyor belt system.Image processing by OCR, Micro-controller, Robotic System, Servomotor,Optical character recognition.

I.INTRODUCTION:

Determining real time and highly accurate characteristics of small objects in a fast flowing stream would open new directions for industrial sorting processes. The present paper relates to an apparatus and method for classify in and sorting small-sized objects, using elect ronic systems and advanced sensors operating on the basis of a physical and geometric characterization of each element. Recent advances in electronics and printed circuit board technology open new perspectives for industrial application in this field. The proposed selection process is based on a multi sensorial characterization, and more specifically on crossed optical and impedimetric analysis of the objects to be sorted. Parallel guides, also called channels, are created on a slanted plant support. The objects to be sorted are immersed in a continuous, free-falling flow along said guides [1] [2].

By another way this project can be treated an automated material handling system & can be designed by following way. It synchronizes the movement of robotic arm to pick the objects moving on a conveyor belt. It aims in classifying the coloured objects which are coming on the conveyor by picking and placing the objects in its respective pre-programmed place. Thereby eliminating the monotonous work done by human, achieving accuracy and speed in the work. The project involves colour sensors that senses the object's colour and sends the signal to the microcontroller. The microcontroller sends signal to circuit which drives the various motors of the robotic arm to grip the object and place it in the specified location. Based upon the colour detected, the robotic arm moves to the specified location, releases the object and comes back to the original position.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

II.SYSTEM MODEL AND ASSUMPTIONS:

The fig. shows block diagram of a system. The basic theme of this project is object flowing on conveyor are sensed, selected and sorted depending on their colour and size. For this, camera is used as input sensor, camera is overhead camera which will be mounted on PC, and will be connected to PC by USB. The camera will take a snap and it will feed to PC for colour processing. In PC matlab is used for processing on colour, depending on this signal will be given to microcontroller Atmega 328.

The microcontroller in turn will control the servomotors by PWM signals. These servomotors will control the movement of robotic arm, by controlling their angular movement. Thus the robotic arm will be fully controlled by servomotors. The gripper of robotic arm will pick the object place it depending on its size. This is full automatic process no manual support is needed. The microcontroller used here is with the support of Arduino kit. The Arduino is good platform for robotics application. It is the software and hardware also, using both the above system is developed. Thus the real time, continuous object sorting can be done.

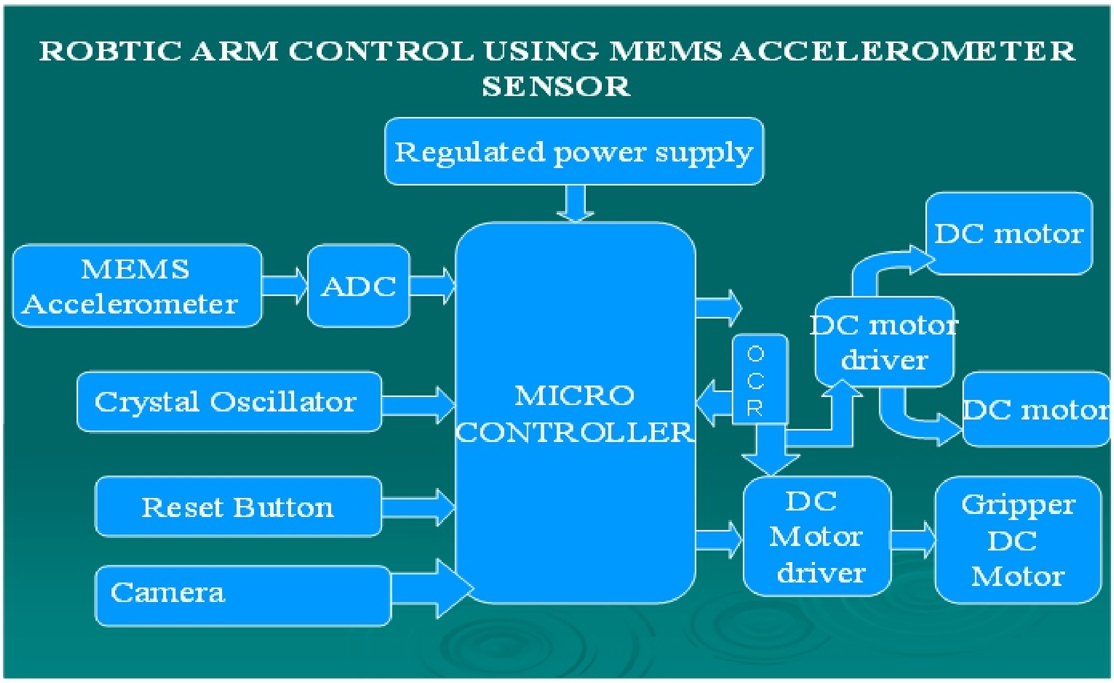


Fig. 1 Block Diagram

A. Microcontroller:

The Arm Raspberry pi is a low-power 32-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, t. 700 MHz ARM1176JZF-S core (ARM11 family, ARMv6 instruction set).The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers. The chip is Broadcom BCM2835 (CPU, GPU, DSP, SDRAM, and single USB port).

RAM is 512 MB.,4usb on board, Storage card is Micro SD and voltage levels are as follows600 mA upto 1.8 A at 5 V.GPIO pins are 40. USB mainly used for key board for peripherals WI-FI Adapter and audio connections using a 3.5 MM Jack SD card is used as a boot device and also persistent storage. More storage can be attached to the USB

B. Camera:

The camera used in this case will be overhead camera, it will take the snapshot of the object for colour sensing purpose. The image captured by the camera will be processed by image processing using Tesseract. The camera used in this case is Logitech PN 960-000748 whose technical specifications are:

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

•Video calling (640 x 480 pixels)
•Video capture: Up to 1024 x 768 pixels
•Fluid Crystal Technology
•Photos: Up to 1.3 megapixels (software enhanced)
•Built-in mic with noise reduction
•Hi-Speed USB 2.0 certified (recommended)
•Universal clip fits laptops, LCD or CRT monitors

Fig. 2 Camera

C. Image Processing using TESSERACT OCR engine:

Tesseract is an open source optical character recognition (OCR) engine originally developed at Hewlett-Packard between 1985 and 1995, but never commercially exploited. It rated highly at The Fourth Annual Test of OCR Accuracy held in 1995 at the University of Nevada, Las Vegas' Information Science However by that time, Tesseract's development had ceased. In 2005, HP transferred Tesseract's unaltered code to the ISRI and it was released as open source. ISRI discovered that the original developer, Ray Smith (see http://research.google.com/pubs/author4479.html), was now employed at Google after several years working on the market leading commercial OCR engine Omnipage. Google were persuaded by ISRI to allow Smith to continue development of Tesseract as open source software. Version 2.0 is now available for download from Google Code at http://code.google.com/p/tesseract-ocr/.he applications.

Limitations of Tesseract:

Tesseract is an OCR engine, not a complete OCR program Tesseract is an OCR engine rather than a fully featured program similar to commercial OCR software such as Nuance's Omnipage. It was originally intended to serve as a component part of other programs or systems. Although Tesseract works from the ommand line, to be usable by the average user the engine must be integrated into other programs or interfaces, such as FreeOCR.net, WeOCR or OCRpous. Without thisintegration into programs such as these, Tesseract has no page layout analysis, no output formatting and no graphical user interface (GUI).

How accurate is Tesseract OCR?

The above processes ensure that Tesseract is highly accurate when recognizing texts from languages that are currently supported. Results from The Fourth Annual Test of OCR Accuracy (example, Tesseract demonstrated a Word Accuracy of 97.69% with a sample of English newspapers. Since these tests, the Tesseract development team at Google claim to have improved Tesseract's general results by 7.31% for2010.

What are the language-specific components of Tesseract?

For a language such as English, 8 components are used: 1. General Words Wordlist (tessdata/eng.word-dawg) 2. Frequent Word Wordlist (tessdata/eng.freq-dawg) 3. User Wordlist (tessdata/eng.user-words) 4. Index for Character Set (tessdata/eng.inttemp5. Box file – for use in locating characters in the training file (tessdata/eng.normproto) 6. Box file – for use in locating characters in the training file (tessdata/eng.pffmtable) 7. Language's Character Set (tessdata/eng.unicharset) 8. Character Cluster Disambiguator - for 'm' and 'rn', for instance. (tessdata/eng.DangAmbigs). OCR technology uses character recognition to attempt to identify the individual characters that make up a printed text. Although the process used to identify individual characters is language independent, Tesseract must be given a list of the specific characters used by a language (item 4 in the list above). Tesseract must then be trained to correctly identify these characters when they appear within a piece of text. Training is done by feeding into Tesseract a document with words, sentences, symbols and numbers from the required language which contains a recommend ten to twenty example of each of the characters used by that language. Such a list has been added to this document as an appendix. This list must be fed in twice, once as digital text and once as a scan of a printed version of the same text. This produces a 'box-file' containing Tesseract's interpretation of the position of characters and their identity. The next part of the process is to manually correct any errors made by Tesseract, for example the identification of ŵ as W or the identification of the letter combination rn as m. A useful utility with a graphical user interface now exists to simplify this process, and is available from the Tesseract project page. Once this task has been finished, common mistakes such as those mentioned above can be added to the Character Cluster Disambiguator file.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

This training process must be repeated with all font types required, including bold, italic and underlined versions of the same font. The Character Cluster Disambiguator file, in conjunction with a language's word list, helps Tesseract identify a word by suggesting possible corrections to certain characters that allow Tesseract to locate the correct word in its word list. For example, the file can be used to suggest to Tesseract that rn, wr, iii, and an could all potentially be misidentifications of the letter m, and Tesseract will search the wordlist accordingly. However, not all languages will have a list of the commonly used words at their disposal. A list of the head words from a dictionary, for example, is not sufficient as all inflected forms must also be included. For example, mouse and mice should both be included in an English wordlist, and so too run and ran. Many other languages undergo far more inflection than English, so their corresponding wordlists are likely to be both longer and harder to create. In Welsh for example, nouns like coffi (coffee) occur regularly as goffi, choffi and choffi, effectively quadrupling the number of nouns in a list. Many European languages have significantly more verbal forms compared with English. This inherent complexity in language is part of the reason that resources such as wordlists have not been develop for many languages with less resources. Bespoke wordlists would have to be created for any language supported where wordlists are not available. In truth, for optimum performance, Tesseract requires not one, but two word lists. One should contain the most frequently used words in a language, which Tesseract will search first, the second, which Tesseract will only search after failing to find a word in the first list, should contain the less frequently used words in a language. A third list for user-added words also exists. In theory, the above steps should allow for the creation of an OCR engine in languages currently unsupported by Tesseract. However, some languages may not be suitable candidates, as right to left languages are currently not compatible with some ofthe hardcoded functionality built into Tesseract. Depending on character sets, some languages with complicated glyphs or characters may also be unsuitable. However, Google are currently working on increased language support in future.

How does Tesseract work?

Outlines are analysed and stored 2. Outlines are gathered together as Blobs 3. Blobs are organized into text lines 4. Text lines are broken into words 5. First pass of recognition process attempts to recognize each word in turn 6. Satisfactory words passed to adaptive trainer7. Lessons learned by adaptive trainer employed in a second pass, which attempts recognize the words that were not recognized satisfactorily in the first pass 8. Fuzzy spaces resolved and text checked for small caps 9. Digital texts are outputted. 7. Lessons learned by adaptive trainer employed in a second pass, which attempts recognize the words that were not recognized satisfactorily in the first pass 8. Fuzzy spaces resolved and text checked for small caps 9. Digital texts are outputted During these processes, Tesseract uses: algorithms for detecting text lines from a skewed page algorithms for detecting proportional and non proportional words (a proportional word is a word where all the letters are the same width) algorithms for chopping joined characters and for associating broken characters linguistic analysis to identify the most likely word formed by a cluster of characters two character classifiers: a static classifier, and an adaptive classifier which employs training data, and which is better at distinguishing between upper and lower case letters.

Line and Word Finding Line Finding:

The line finding algorithm is one of the few parts of Tesseract that has previously been published [3]. The line finding algorithm is designed so that a skewed page can be recognized without having to de-skew, thus saving loss of image quality. The key parts of the process are blob filtering and line construction. Assuming that page layout analysis has already provided text regions of a roughly uniform text size, a simple percentile height filter removes drop-caps and vertically touching characters. The median height approximates the text size in the region, so it is safe to filter out blobs that are smaller than some fraction of the median height, being most likely punctuation, diacritical marks and noise. Estimate the baselines, the filtered blobs are more likely to fit a model of non-overlapping, parallel, but sloping lines. Sorting and processing the blobs by x-coordinate makes it possible to assign blobs to a unique text line, while tracking the slope across the page, with greatly reduced danger of assigning to an incorrect text line in the presence of skew. Once the filtered blobs have been assigned to lines, a least median of squares fit [4] is used to estimate the baselines, and the filtered-out blobs are fitted back into the appropriate lines. The final step of the line creation process merges blobs that overlap by at least half horizontally, putting diacritical marks together with the correct base and correctly associating parts of some broken characters.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Baseline Fitting:

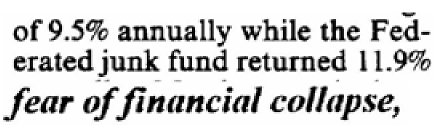
Once the text lines have been found, the baselines are fitted more precisely using a quadratic spline. This Fixed Pitch Detection and Chopping Tesseract tests the text lines to determine whether they are fixed pitch. Where it finds fixed pitch text, Tesseract chops the words into characters using the pitch, and disables the chopper and associator on these words for the word recognition step.

Fixed Pitch Detection and Chopping:

Tesseract tests the text lines to determine whether they are fixed pitch. Where it finds fixed pitch text, Tesseract chops the words into characters using the pitch, and disables the chopper and associate on these words for the word recognition step.

Proportional Word Finding Non-fixed-pitch or proportional text spacing:

The gap between the tens and units of '11.9%' is a similar size to the general space, and is certainly larger than the kerned space between 'erated' and 'junk'. There is no horizontal gap at all between the bounding boxes of 'of' and 'financial'. Tesseract solves most of these problems by measuring gaps in a limited vertical range between the baseline and mean line. Spaces that are close to the threshold at this stage are made fuzzy, so that a final decision can be made after word recognition.



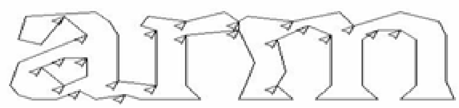
Word Recognition:

Part of the recognition process for any character recognition engine is to identify how a word should be segmented into characters. The initial segmentation output from line finding is classified first.

The rest of the word recognition step applies only to non-fixed-pitch text.Arduino is a tool for making computers that can sense and control more of the physical world than your desktop computer. It's an open-source physical computing platform based on a simple

Chopping Joined Characters:

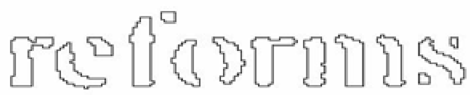
While the result from a word (see section 6) is unsatisfactory, Tesseract attempts to improve the result by chopping the blob with worst confidence from the character classifier. Candidate chop points are found from concave vertices of a polygonal approximation [2] of the outline, and may have either another concave vertex opposite, or a line segment. It may take up to 3 pairs of chop points to successfully separate joined characters from the ASCII set. Fig. 4 shows a set of candidate chop points with arrows and the selected chop as a line across the outline where the 'r' touches the 'm'.Chops are executed in priority order. Any chop that fails to improve the confidence of the result is undone, but not completely discarded so that the chop can be re-used later by the associator if needed.



Associating Broken Characters:

When the potential chops have been exhausted, if the word is still not good enough, it is given to the associator. The associator makes an A* (best first) search of the segmentation graph of possible combinations of the maximally chopped blobs into candidate characters. It does this without actually building the segmentation graph, but instead maintains a hash table of visited states.The A* search proceeds by pulling candidate new states from a priority queue and evaluating them by classifying unclassified combinations of fragments. It may be argued that this fully-chop-then-associate approach is at best inefficient, at worst liable to miss important chops, and that may well be the case. The advantage is that the chop-then-associate scheme simplifies the data structures that would be required to maintain the full segmentation graph. When the A* segmentation search was first implemented in about 1989, Tesseract's accuracy on broken characters was well ahead of the commercial engines of the day. Fig. 5 is a typical example. An essential part of that success was the character classifier that could easily recognize broken characters.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal



Static Character Classifier:

An early version of Tesseract used topological features developed from the work of Shillman et. al. [7-8] Though nicely independent of font and size, these features are not robust to the problems found in real-life images, as Bokser [9] describes. An intermediate idea involved the use of segments of the polygonal approximation as features, but this approach is also not robust to damaged characters. For example, in Fig. 6(a), the right side of the shaft is in two main pieces, but in Fig. 6(b) there is just a single piece.The breakthrough solution is the idea that the features in the unknown need not be the same as the features in the training data. During training, the segments of a polygonal approximation [2] are used for features, but in recognition, features of a small, fixed length (in normalized units) are extracted from the outline and matched many-to-one against the clustered prototype features of the training data. In Fig. 6(c), the short, thick lines are the features extracted from the unknown, and the thin, longer lines are the clustered segments of the polygonal approximation that are used as prototypes. One prototype bridging the two pieces is completely unmatched. Three features on one side and two on the other are unmatched, but, apart from those, every prototype and every feature is well matched. This example shows that this process of small features matching large prototypes is easily able to cope with recognition of damaged images. Its main problem is that the computational cost of computing the distance between an unknown and a prototype is very high.

Linguistic Analysis:

Tesseract contains relatively little linguistic analysis. Whenever the word recognition module is considering a new segmentation, the linguistic module (mis-named the permuter) chooses the best available word string in each of the following categories: Top frequent word, Top dictionary word, Top numeric word, Top UPPER case word, Top lower case word (with optional initial upper), Top classifier choice. word. The final decision for a given segmentation is simply the word with the lowest total distance rating, where each of the above categories is multiplied by a different constant.

Words from different segmentations may have different numbers of characters in them. It is hard to compare these words directly, even where a classifier claims to be producing probabilities, which Tesseract does not. This problem is solved in Tesseract by generating two numbers for each character classification. The first, called the confidence, is minus the normalized distance from the prototype. This enables it to be a "confidence" in the sense that greater numbers are better, but still a distance, as, the farther from zero, the greater the distance. The second output, called the rating, multiplies the normalized distance from the prototype by the total outline length in the unknown character. Ratings for characters within a word can be summed meaningfully, since the total outline length for all characters within a word is always the same.

Adaptive Classifier:

It has been suggested [11] and demonstrated [12] that OCR engines can benefit from the use of an adaptive classifier. Since the static classifier has to be good at generalizing to any kind of font, its ability to discriminate between different characters or between characters and non-characters is weakened. A more font-sensitive adaptive classifier that is trained by the output of the static classifier is therefore commonly [13] used to obtain greater discrimination within each document, where the number of fonts is limited.

Tesseract does not employ a template classifier, but uses the same features and classifier as the static classifier. The only significant difference between the static classifier and the adaptive classifier, apart from the training data, is that the adaptive classifier uses isotropic baseline/x-height normalization, whereas the static classifier normalizes characters by the centroid (first moments) for position and second moments for anisotropic size normalization.

The baseline/x-height normalization makes it easier to distinguish upper and lower case characters as well as improving immunity to noise specks. The main benefit of character moment normalization is removal of font aspect ratio and some degree of font stroke width. It also makes recognition of sub and superscripts simpler, but requires an additional classifier feature to distinguish some upper and lower case characters. Fig. 7 shows an example of 3 letters in baseline/x-height normalized form and moment normalized form.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

E. Robotic Arms & Servomotors:

Arms are types of jointed robot manipulator that allow robots to interact with their environment. Many have onboard controllers or translators to simplify communication, though they may be controlled directly or in any number of ways. Due to this fact, standalone arms are often classified as full robots. The robot used in this project is 4 Axis Robotic Arm. 4 Axis Robotic Arm is designed for small mobile robots. It can grip objects with the size up to 60mm with the force up to 250gms. Arm has reach of 23cm. It can lift the payload up to 400gms. Robotic Arm comes fully assembled and ready to use. First two axis of the arm are made up of NRS-995 dual bearing heavy duty metal gear motors and remaining 2 axis and gripper uses NRS-585 dual bearing plastic gear servo motors. Axis 2 and 3 enables gripper to maintain its angle constant with the surface while moving up and down. Robotic arm can do Left-Right, Up-Down while keeping gripper parallel to surface, Twist motions and Gripping action. Robotic Arm will require current up to 5Amps. Make sure that your robot can supply that much amount of current for proper operation of the arm. The robotic arm has following specifications.

Number of Axis: 4 + Gripper

Gripping force: 250gms (Maximum) Gripping jaw length: 43mm

Gripping jaw width: 60mm

Weight: 541gms (Including 2 NRS-995 and 3 NRS-585 servo motors)

Operating voltage: 5V to 6V Reach: 23cm

Servos are DC motors with built in gearing and feedback control loop circuitry. And no motor drivers required. A servomotor is a rotary actuator that allows for precise control of angular position. They consist of a motor coupled to a sensor for position feedback, through a reduction gearbox. They also require a relatively sophisticated controller, often a dedicated module designed specifically for use with servomotors. Servomotors are used in applications such as robotics, CNC machinery or automated manufacturing. The servo motor has some control circuits and a potentiometer (a variable resistor) that is connected to the output shaft. This pot allows the control circuitry to monitor the current angle of the servo motor. If the shaft is at the correct angle, then the motor shuts off. If the circuit finds that the angle is not correct, it will turn the motor the correct direction until the angle is correct. The output shaft of the servo is capableof traveling somewhere around 180 degrees.Usually, its somewhere in the 210 degree range, but it varies by manufacturer. A normal servo is used to control an angular motion of between 0 and 180 degrees.

Table I. Axis Capabilities:

Axis Capabilities: Mechanical Assembly	Maximum Angle(º)	Speed (Degree/sec)
Waist	180º	0-27º
First Arm	180º	0-27º
Second Arm	180º	0-27º
Third Arm	180º	0-27º
Forth Arm	180º	0-27º

A normal servo is mechanically not capable of turning any farther due to a mechanical stop built on to the main output gear. The amount of power applied to the motor is proportional to the distance it needs to travel. So, if the shaft needs to turn a large distance, the motor will run at full speed. If it needs to turn only a small amount, the motor will run at a slower speed.The motor is paired with some type of encoder to provide position and speed feedback. In the simplest case, only the position is measured. The measured position of the output is compared to the command position, the external input to the controller. If the output position differs from that required, an error signal is generated which then causes the motor to rotate in either direction, as needed to bring the output shaft to the appropriate position.

As the positions approach, the error signal reduces to zero and the motor stops.More sophisticated servomotors measure both the position and also the speed of the output shaft. They may also control the speed of their motor, rather than always running at full speed. Both of these enhancements, usually in combination with a PID control algorithm, allow the servomotor to be brought to its commanded position more quickly and more precisely, with less overshooting. The servo turn rate, or transit time, is used for determining servo rotational velocity. This is the amount of time it takes for the servo to move a set amount, usually 60 degrees. For example, suppose you have a servo with a transit time of 0.17sec/60 degrees at no load, this means it would take nearly half a second to rotate an entire 180 degrees

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

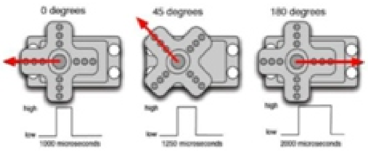


Fig.5 Servomotor Rotation

G. Conveyor Belt:

The conveyor motor receives power from battery. A conveyor belt consists of two or more pulleys, with a continuous loop of material - the conveyor belt - that rotates about them. One or both of the pulleys are powered, moving the belt and the material on the belt forward. The powered pulley is called the drive pulley while the unpowered pulley is called the idler. Conveyor frames are supplied with either butting plate (standard) or hook and bar attachments to secure each segment together. Heavy duty rollers are supplied with shafts.

Fig. 5Conveyor Belt

III. Result

We can assume objects in circular, rectangular shape in different colours so the result is

Table II. Result:

Sr. No.	colour	No. of object	Result	
			Sorted	Not Sorted
1	Red	15	12	3
2	BLUE	8	5	3
3	GREEN	5	3	1
4	Orange	5	2	3
5	Navy Blue	3	2	1

Tesseract was included in the 4th UNLV annual test [1] of OCR accuracy, as "HP Labs OCR," but the code has changed a lot since then, including conversion to Unicode and retraining. Table 1 compares results from a recent version of Tesseract (shown as 2.0) with the original 1995 results (shown as HP). All four 300 DPI binary test sets that were used in the 1995 test are shown, along with the number of errors (Errs), the percent error rate (%Err) and the percent change relative to the 1995 results (%Chg) for both character errors and non-stopword errors. [1] More up-to-date results are at http://code.google.com/p/tesseract-ocr.

		Character			Word		
Ver	Set	Errs	%Err	%Chg	Errs	%Err	%Chg
HP	bus	5959	1.86		1293	4.27	
2.0	bus	6449	2.02	8.22	1295	4.28	0.15
HP	doe	36349	2.48		7042	5.13	
2.0	doe	29921	2.04	-17.68	6791	4.95	-3.56
HP	mag	15043	2.26		3379	5.01	
2.0	mag	14814	2.22	-1.52	3133	4.64	-7.28
HP	news	6432	1.31		1502	3.06	
2.0	news	7935	1.61	23.36	1284	2.62	-14.51
2.0	total	59119		-7.31	12503		-5.39

IV. CONCLUSION AND FURTHER WORK:

Fully functional sorter machine can be implemented by using a structure of parallel and independent channels in order to increase the overall throughput which results with a forecasted performance. The project can work successfully. There are two main steps in sensing part, objects detection and recognition. The system can successfully perform handling station task, namely pick and place mechanism with help of sensor. Thus a cost effective Mechatronics system can be designed using the simplest concepts and efficient result can be observed.After lying dormant for more than 10 years, Tesseract is now behind the leading commercial engines in terms of its accuracy. Its key strength is probably its unusual choice of features. Its key weakness is probably its use of a polygonal approximation as input to the classifier instead of the raw outlines.With internationalization done, accuracy could probably be improved significantly with the judicious addition of a Hidden-Markov-Model-based character n-gram model,

Acknowledgements:

The author would like to thank John Burns and Tom Nartker for their efforts in making Tesseract open source, the ISRI group at UNLV for sharing their tools and data, as well as Luc Vincent, Igor Krivokon, Dar-Shyang Lee,

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

and Thomas Kielbus for their comments on the content of this paper.

REFERENCES:

J.S.V. Rice, F.R. Jenkins, T.A. Nartker, The Fourth Annual Test of OCR Accuracy, Technical Report 95-03, Information Science Research Institute, University of Nevada, Las Vegas, July 1995.
K.R.W. Smith, The Extraction and Recognition of Text from Multimedia Document Images, PhD Thesis, University of Bristol, November 1987.
L R. Smith, "A Simple and Efficient Skew Detection Algorithm via Text Row Accumulation", Proc. of the 3rd Int. Conf. on Document Analysis and Recognition (Vol. 2), IEEE 1995, pp. 1145-1148.
M.P.J. Rousseeuw, A.M. Leroy, Robust Regression and Outlier Detection, Wiley-IEEE, 2003.
N.S.V. Rice, G. Nagy, T.A. Nartker, Optical Character Recognition: An Illustrated Guide to the Frontier, Kluwer Academic Publishers, USA 1999, pp. 57-60.
O.P.J. Schneider, "An Algorithm for Automatically Fitting Digitized Curves", in A.S. Glassner, Graphics Gems I, Morgan Kaufmann, 1990, pp. 612-626.
P.R.J. Shillman, Character Recognition Based on Phenomenological Attributes: Theory and Methods, PhD. Thesis, Massachusetts Institute of Technology. 1974.
Q.B.A. Blesser, T.T. Kuklinski, R.J. Shillman, "Empirical Tests for Feature Selection Based on a Pscychological Theory of Character Recognition", Pattern Recognition 8(2), Elsevier, New York, 1976.
R M. Bokser, "Omnidocument Technologies", Proc. IEEE 80(7), IEEE, USA, Jul 1992, pp. 1066-1078.
S.H.S. Baird, R. Fossey, "A 100-Font Classifier", Proc. of the 1st Int. Conf. on Document Analysis and Recognition, IEEE, 1991, pp 332-340.
T.G. Nagy, "At the frontiers of OCR", Proc. IEEE 80(7), IEEE, USA, Jul 1992, pp 1093-1100.
U.G. Nagy, Y. Xu, "Automatic Prototype Extraction for Adaptive OCR", Proc. of the 4th Int. Conf. on Document Analysis and Recognition, IEEE, Aug 1997, pp 278-282.
V.I. Marosi, "Industrial OCR approaches: architecture, algorithms and adaptation techniques", Document Recognition and Retrieval XIV, SPIE Jan 2007, 6500-01.
[1]AlessandroGolfarelli, RossanoCodeluppi and Marco Tartagni, "A Self-Learning Multi-Sensing Selection Process: Measuring Objects One by One by",ARCES – LYRAS LAB University of Bologna, Campus of Forlì, 1-4244-1262-5/07/$25.00 ©2007 IEEE, IEEE SENSORS 2007 Conference.
[2]Sahu, S., Lenka, P.; Kumari, S.; Sahu, K.B.; Mallick, B.; "Design a colour sensor: Application to robot handling radiation work", Vol. 56, No. 10, pp. 365- 368, 2007, Industrial. Engineering.
[3]www.google.com/microepsilon.com/catcolorsensor—e
[4]www.shortcourse.com/www.sensors.com/optical/imagesens ors
[5]wwwgoogle.com/wisegeek.com/whatis optical sensors.htmwww.google.com/osa.org/sensors
[6]www.pdfgenicom/compacsort.com/sorters
[7]www.pdfgenicom/indiamart.com/sorting machine

Author's Profile

Mr. SWARNENDU SARKAR pursuing M.Tech (ECE) from St.Martins Engineering College, JNTUH Hyderabad, and B.Tech (ECE) from CM Engineering College,JNTUH Hyderabad. He is working as Embedded Engineer in Ritchie Technocrats, R.R Dist, AP and India.

Mr. K.YADAIAH, pursuing PhD in JNTUH under the guidance of Dr. B.L. RAJU, Principal ACE Engineering College. Post Graduated in Electronics and communication Engineering (M.Tech) From JNTUCE, Kakinada in Nov-2003 and Graduated in ECE (B.Tech) from JNTU, Hyderabad in 2001. He is working as An Associate Professor and Head, Department of ECE in St.Martins Engineering College, R.R Dist, AP and India. He has 13+ years of Teaching Experience. His Research area in Ph.D is Energy Efficient Routing in MANETS & Wireless Sensor Networks.