

## Research Article

# Intelligent Machine Vision Based Modeling and Positioning System in Sand Casting Process

**Shahid Ikramullah Butt,<sup>1</sup> Umer Asgher,<sup>1</sup> Umar Mushtaq,<sup>1</sup> Riaz Ahmed,<sup>2</sup> Faping Zhang,<sup>3</sup> Yasar Ayaz,<sup>1</sup> Mohsin Jamil,<sup>1</sup> and Muhammad Kamal Amjad<sup>1</sup>**

<sup>1</sup>School of Mechanical & Manufacturing Engineering, National University of Sciences and Technology (NUST), Islamabad, Pakistan

<sup>2</sup>Directorate of Quality Assurance, National University of Sciences and Technology (NUST), Islamabad, Pakistan

<sup>3</sup>School of Mechanical Engineering, Beijing Institution of Technology, Beijing, China

Correspondence should be addressed to Shahid Ikramullah Butt; drshahid@smme.nust.edu.pk

Received 21 October 2016; Revised 31 December 2016; Accepted 16 January 2017; Published 8 February 2017

Academic Editor: Charles C. Sorrell

Copyright © 2017 Shahid Ikramullah Butt et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Advanced vision solutions enable manufacturers in the technology sector to reconcile both competitive and regulatory concerns and address the need for immaculate fault detection and quality assurance. The modern manufacturing has completely shifted from the manual inspections to the machine assisted vision inspection methodology. Furthermore, the research outcomes in industrial automation have revolutionized the whole product development strategy. The purpose of this research paper is to introduce a new scheme of automation in the sand casting process by means of machine vision based technology for mold positioning. Automation has been achieved by developing a novel system in which casting molds of different sizes, having different pouring cup location and radius, position themselves in front of the induction furnace such that the center of pouring cup comes directly beneath the pouring point of furnace. The coordinates of the center of pouring cup are found by using computer vision algorithms. The output is then transferred to a microcontroller which controls the alignment mechanism on which the mold is placed at the optimum location.

## 1. Introduction

The casting process is used for forming metals into the desired shapes and sizes. It involves pouring boiling hot metal from a ladle into a mold by the workers. Handling molten metals at such a high temperature is an extremely delicate task and requires a lot of skill. Accidents occurring during the casting process can result in severe injuries or even death of the skilled workers. Therefore, a lot of research has been done to introduce automation in the casting process to improve the safety at the working environment and ergonomics. Some of the efforts in this area can be seen in the work by different scholars [1, 2]. In this paper, a novel scheme is presented for introducing automation in the sand casting process. There may be two categories of automation in sand casting process as follows:

(i) The automation of the metal pouring, where a lot of research and development has already been done in this area.

(ii) To develop such a system in which casting molds of different sizes, having different pouring cup location and radius, position themselves in front of the induction furnace such that the center of pouring cup comes directly beneath the pouring point of furnace (very latest research is being carried out in this area).

The idea of sand casting automation is a new concept and the same has been successfully implemented in this research project. The process includes a vision based system, in which a single camera takes the image of the casting mold, Harris Corner Detector technique is used for detecting a reference point on the pallet (on which the mold is placed), then the pouring cup center is found using Hough Transform, explained in [3–5], and the data is transferred to the controller which controls the alignment mechanism of the mold. To verify the concept a small scale model of the system has also been developed; it consists of three main parts which include design and fabrication of the mechanical model, design and fabrication of electronics hardware, and vision

based algorithm and microcontroller algorithm. All the parts of the small scale model have been successfully completed and integrated to produce an automated mold positioning system.

The rest of the paper has been arranged as follows. Section 2 deals with the implemented vision system methodology, Section 3 addresses the design of the test platform, and Section 4 describes experimental results and analysis. Discussion and conclusions are drawn in Section 5.

Many studies have been conducted to introduce automation in the process of casting. Researchers [6, 7] have worked on a teaching/playback method to ensure that molten metal is quickly and precisely poured into a mold and does not spill over the sprue cup. The filling weight control has been implemented by using the state estimation of the pouring process and the predictive sequence control. Furthermore, to realize highly accurate flow rate control, a feed forward control employing the inverse dynamics of the flow rate model is also proposed [8]. Similarly, the presented Melpore system used teaching/playback method to automate the pouring system of the tilting-ladle type pouring system applying fuzzy control to the flow rate control system in tilting-ladle type automatic pouring system. Noda et al. [9] have optimized the sequence control of automatic pouring system in press casting process; recently modern manufacturing and automation [10, 11] have achieved the high precision positioning control of the pouring liquid while avoiding contact with the mold. However, all these techniques focus on the improvement and automation of the metal pouring. The problem of high precision positioning of molds of different sizes and having different pouring cup locations has not been addressed so far. In computer and machine vision's image capturing techniques, image is often corrupted while acquisition and some techniques like Normalized Convolution (NC) and High Resolution Normalized Convolution (HR-NC) are applied on the corrupted image in order to regain the missing and corrupted minor details which are lost in data acquisition [12].

## 2. Methodology and Experimental Setup

**2.1. Machine Vision System, Its Alignment, Feedback, and Robot Guidance.** Machine vision systems are being deployed as a mandatory requirement worldwide to ensure quality products. In our setup, need of four cameras which are generally used in the system to capture the complete surface of the product and to generate a composite surface is eliminated. Instead, each image is individually processed and if any individual part fails the inspection, it is tracked and rejected at the end of the process. To ensure that the top of the product maintains critical dimensions, the vertically positioned camera obtains an image of the top of the product. Using Mil measurement (The American System of Measures used in manufacturing dimensions like distance and length, etc.) for thickness functions, the inner and outer circumferences of the top can be located and used to plot the circumference. After this, features such as inner diameter, outer diameter, and ovality can be calculated and compared with known good measurements [10]. 2D representation of the system is shown in Figure 1.

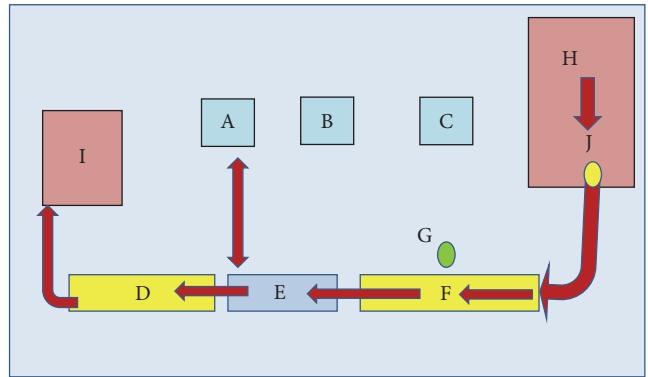


FIGURE 1: 2D representation of system, where A is induction furnace, B is induction heating generator, C is transformer, D is conveyor 1, E is conveyor 2, F is conveyor 3, G is camera, H is mold preparation area, I is cooling area, and J is mold and pallet.

The process starts from H which is the mold preparation area; inputs to this section are the raw material for mold preparation (sand, binder), mold parts (cope, drag), and pattern. Using these, molds are prepared and stacked. J represents a final prepared mold ready to enter the conveyor system. F is a transport conveyor which takes the mold from the preparation area to the module E. Within conveyor 3 a camera is mounted, the mold is detected with the help of proximity sensors, and when it is under the camera the conveyor stops for a moment while the camera takes an image of the mold, after which the conveyor restarts. The picture taken by the camera is sent to the computer where it undergoes an image processing algorithm which tells exactly the location in X and Y coordinates of the center of the pouring cup. This information will be used later to align the mold beneath the pouring point of the furnace. After passing the camera, mold reaches the end of conveyor 3, where it enters conveyor 2. The distance it travels on conveyor 2 is given by the controller from the output of the image processing algorithm; therefore in Y-direction the pouring cup center is aligned to the pouring point of furnace; now conveyor 2 moves forward by the screw mechanism on its base towards the furnace and the distance it moves is again given by the controller from the output of the image processing algorithm; therefore now the mold gets aligned in X-direction also, so when conveyor 2 stops at the furnace, the pouring cup center will be exactly beneath the pouring point of furnace. When the operator gets the signal that the mold is aligned, he manually pours the metal in the mold by simply rotating the handle on the induction furnace which tilts the crucible forward; after pouring is done the operator sends a signal to the system and conveyor 2 moves back in X-direction. After it reaches the end position, the mold moves from conveyor 2 to conveyor 1, and when conveyor 1 gets fully stacked, the operator shifts the molds to the cooling area (I). After molds are cooled, they are broken and the parts are retrieved while the sand, cope, and drag are sent back to the mold preparation area. Thus the same cycle repeats again and again.

When assembling components into a larger product, it is often necessary to align one part with another prior to



FIGURE 2: Model of casting mold on pallet.

insertion or attachment in some way. Machine vision systems can easily provide the alignment systems with the positional feedback they require to make sure the components fit together in the proper manner as shown in Figure 1. The algorithm for vision system has been developed in MATLAB, which is interfaced to the external microcontroller based hardware; thus the output of the vision system acts as an input to the microcontroller system. The aim of the vision system is to find the location/coordinates of the center of pouring cup of the casting mold (Figures 2 and 6). The mold can be of different sizes, with different positions and orientations on the pallet, and can have different position and radius pouring cup. There are three main sections of the image processing algorithm as presented in the succeeding sections.

**2.2. Circular Hough Transform: Part Presence and Orientation.** The pouring cup of a sand casting mold is actually a circular hole; thus the Circular Hough Transform [11] technique is implemented to detect the coordinates of the center of this hole. The steps performed in the developed algorithm are given below.

Convert RGB image to grey scale:

- Define and apply averaging filter.
- Apply canny edge detector, giving an output in the form of binary image, shown in Figure 2.
- Define a 3-dimensional accumulator array, with 2 dimensions representing the rows and columns of original image and the third dimension representing the range of radius of circle to look for.
- Set the 3D accumulator array to zero.
- Scan entire image (from step (c)):
  - If edge is detected (i.e., a value of 1 is found), proceed to next step; else keep scanning.
  - Define limits to draw a circle in the accumulator array.
  - Draw a circle in the accumulator array centered at the index of the edge point detected.
  - Draw circles in the accumulator array shown in Figure 3 for all the values of radius defined earlier in the accumulator array, centered at the index of the edge detected; this is also called voting (drawing a circle means to add 1 to the corresponding array elements). The modern

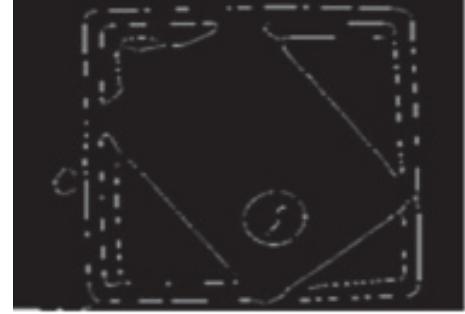


FIGURE 3: Output of canny edge detector.

research on machine vision for tolerance detection [13] suggests novel methods to detect the dimensions with accuracy and we tried to implement our algorithm on similar lines. This system (as shown in Figures 4(a) and 4(b)) is designed such that the pouring cup of any radius should be accommodated. To detect the radius of circle, we draw circles on the accumulator array of that known diameter; if the diameter of circle is unknown, then for that purpose some modifications need to be done in the existing algorithm. Firstly, the accumulator array is of three dimensions, previously it was 2D (rows and columns), and now the third dimension in the array is of the radius. As the dimension has to be finite, so for the algorithm to complete its work a range of radius needs to be given to look for in the image. This range can be as big as we desire, but the bigger it gets the more computational power and time it would take to achieve the output; thus it is better to give an optimized range of radius of which the pouring cup can exist. Now, instead of drawing circles in the accumulator array we draw cones, with the starting radius of the cone as the staring value given in the third dimension of the array and the ending radius as the maximum value given in the 3D array. To simplify, it can also be looked as a stack of 2D arrays with the radius incrementing with each 2D array; all the arrays stacked on top of each other will give a cone. Now, the point where all the cones intersect will have the maximum magnitude and thus similarly scanning the 3D array for the index of maximum value will give us the location of the center point of the pouring cup. In this case one more additional piece of information will be retrieved; when the scanning of the 3D array will give output of the index of the maximum value pixel, it will contain three values as it is a 3D array, first two will be defining the position (columns and rows), and the third would be radius; thus in this we can also tell the radius of the detected circle along with the coordinated of the center.

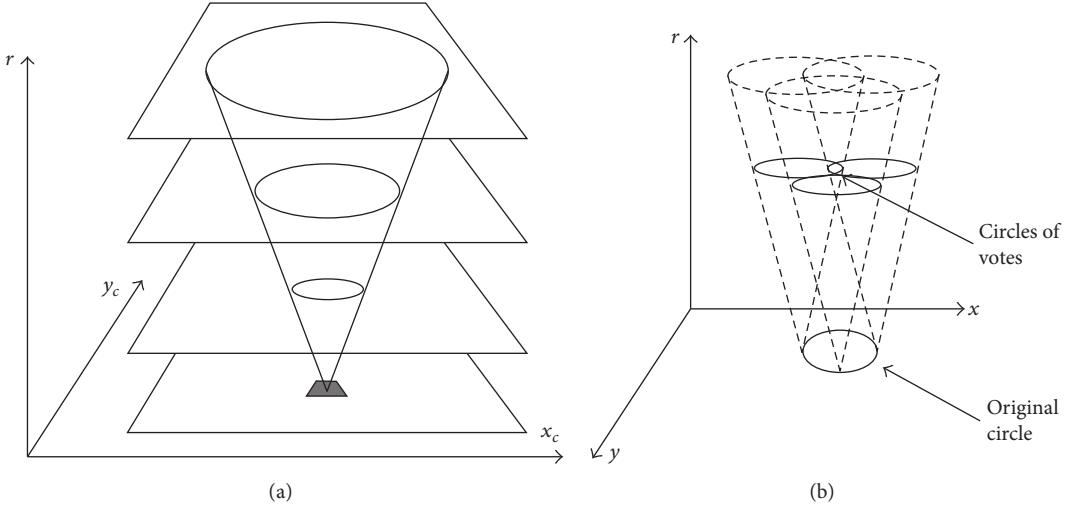


FIGURE 4: (a) 3D accumulator array in planner form. (b) 3D accumulator array in multiconical form.

- (f) After complete image has been scanned and the accumulator array has been filled, scan the array for the maximum value.
- (g) The index of the max value tells the coordinates of the center of actual circle detected in rows and columns, and as the index has 3 values, it also tells the radius of detected circle.
- (h) The coordinates found are in pixels right now and will later be used to find the location of pouring cup center in millimeters.

**2.3. Harris Corner Detectors.** Finding just the coordinates of the pouring cup center is not enough for mold positioning. A reference point is needed from here; center measured is selected from the top right corner of the pallet. Harris Corner Detector technique [11, 14] is used to detect and locate the corner. The following is the final equation on which the algorithm is developed:

$$E(u, v) = (u, v) M \begin{pmatrix} u \\ v \end{pmatrix} \quad (1)$$

$$M = \sum_{x,y} w(x, y) \begin{pmatrix} I_x \cdot I_x & I_x \cdot I_y \\ I_x \cdot I_y & I_y \cdot I_y \end{pmatrix},$$

where

$E(u, v)$  is change in intensity by moving a filter window over the image,  
 $u$  and  $v$  are shifts in the window.

An extended form of General Hough Transform, Circular Hough Transform (CHT), is used to detect circles. In this the “Image” is the input image, “Output” is an edge detected binary image, “thresh” is a two-element vector in which the first element is the low threshold, and the second element is the high threshold, and sigma is the standard deviation of the Gaussian filter. Similar image processing techniques are also

used in food processing industry for detection of dimension and quality [15]. In CHT, we first initiate an accumulator array which is of the same size as the edge detected image. Then this image is scanned and at each point where an edge is detected it votes for a possibility of a circle in the accumulator array. Let us first take example where the radius of circle to be detected is known and is constant. Figure 5(a) is an ideal edge detected image from the canny detector and Figure 5(b) is the accumulator array of the same size. Initially the accumulator array is initialized to zero. The binary image is scanned and wherever an edge is detected some operation is done on the accumulator array. An edge is simply a point where the value of pixel is 1 and 0 (zero) where there is no edge. During scanning point 1 is detected as an edge; now the index of point 1 (the row and column position where it lies) is noted and on the same index on the accumulator array a circle is drawn with the center point of the circle on that index. Drawing a circle basically means adding 1 to the pixels of the circle boundary/circumference. Now, similarly point 2 and point 3 will also be detected as an edge and same operation will be done on the accumulator array as done due to point 1. Over here as an example only 3 points are shown, while during scanning each point of the circle edge will be detected and the same operation as of point 1 explained will be done in the accumulator array. As each circle drawn in the accumulator array adds 1 to the circle boundary, then there is a point in the array which will have a maximum magnitude; this is the point where all the drawn circles intersect. As each drawn circle passes through this intersection point and each circle votes/adds 1 to this point, in the end this intersection point has the maximum magnitude. If you notice, now this intersection point of all drawn circles is basically on the same index as that of the center point of the actual circle in the original image. Therefore, after the accumulator has been completely filled we just scan it to find the index of the pixel with maximum magnitude and that index is basically the center of the circle we were looking for.

The Eigen values of matrix  $M$  can help us locate the possible corners in an image. The steps occurring in the

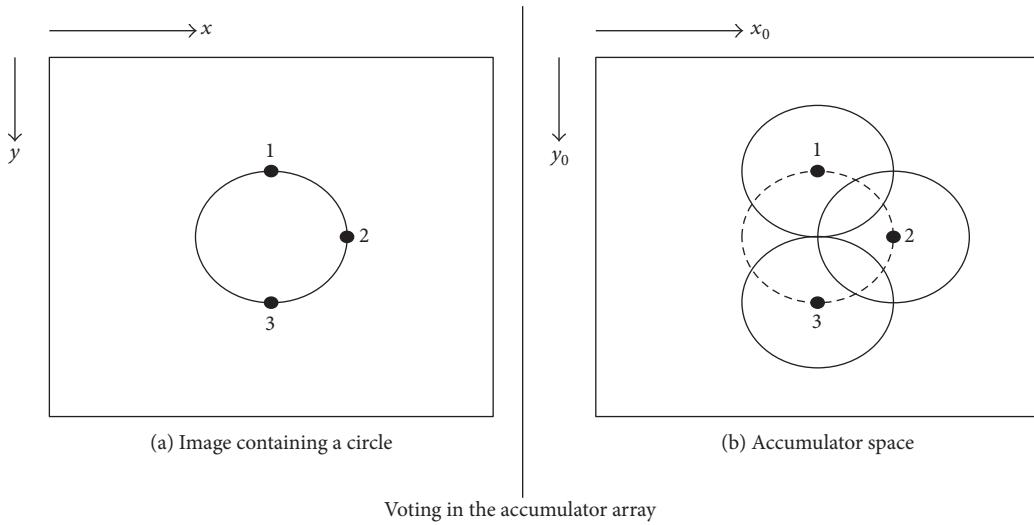


FIGURE 5: (a) Image containing circle. (b) Accumulator space.

developed algorithm are as follows:

(a) Define and apply derivative mask

$$Hx = [-1 \ 0 \ 1; -2 \ 0 \ 2; -1 \ 0 \ 1] \quad (2)$$

on image in  $x$ -direction to get  $Ix$ .

(b) Define and apply derivative mask  $Hy$  ( $Hy$  = transpose of  $Hx$ ) on image in  $y$ -direction to get  $Iy$ .

(c) From  $Ix$  and  $Iy$  calculate  $Ix \cdot Ix$ ,  $Iy \cdot Iy$ , and  $Ix \cdot Iy$ .

(d) Convert original color image to gray scale.

(e) Apply Gaussian filter to image obtained from step (d).

(f) Initiate corner detection array ( $R$ ) to zeros with size of  $R$  equal to the size of original image.

(g) Scan entire image from step (e):

(i) For each pixel define a  $2 \times 2$  matrix

$$M = [Ix \cdot Ix(m, n) \ Ixy(m, n); \ Ixy(m, n) \ Iy \cdot Iy(m, n)], \quad (3)$$

where  $m$  and  $n$  are indices of the specific pixel.

(ii) Find the two Eigen values of matrix  $M$ , which are  $E1$  and  $E2$ .

(iii) Use  $E1$  and  $E2$  in the formula  $r = (E1 * E2)/(E1 + E2)$ .

(iv) Add the  $r$  value obtained to the matrix  $R$  on the same index as of the pixel for which matrix  $M$  was formed in the original image.

(v) Repeat steps (i)–(iv) for each pixel in the image, to get a completely filled matrix  $R$  (finally matrix  $R$  is filled with  $r$  values).

(h) Apply nonmaxima suppression to the matrix  $R$ ; this eliminates the possibility of detecting a specific corner more than once. (In this, the neighborhood of a corner is scanned and the pixel with the largest magnitude which represents the strongest corner position is left while others in its neighborhood with lesser magnitudes are erased.)

(i) Scan matrix  $R$  from step (h), using a threshold which dictates how strong corner is required.

(j) Scan locally matrix  $R$  to find the coordinate of the required reference point. The output of this algorithm is coordinates  $(x_2, y_2)$  in pixels of the top right corner of the pallet. Now this corner coordinates and the pouring cup center coordinates  $(x_1, y_1)$  found earlier will be used to find the distance (shown in Figure 6) in millimeters of the pouring cup center from the reference point.

**2.4. Scaling.** Scaling is used to convert image distance into spatial distance, that is, in millimeter; there can be two approaches to it. First is to use a formula shown below:

$$D_m = \frac{D_i \cdot D_c}{f}, \quad (4)$$

where

$D_m$  is distance required in millimeter (mm),

$D_i$  is distance in the image (in this case it is the distance between the two points shown in Figure 4),

$D_c$  is distance between the camera and mold surface, and  $f$  is the focal length of the camera.

If the focal length of camera is unknown, a practical way can be adopted to formulate the same relationship by finding the constant value  $(D_c/f)$ . One side of the pallet is measured physically in millimeter; then the same side is measured in

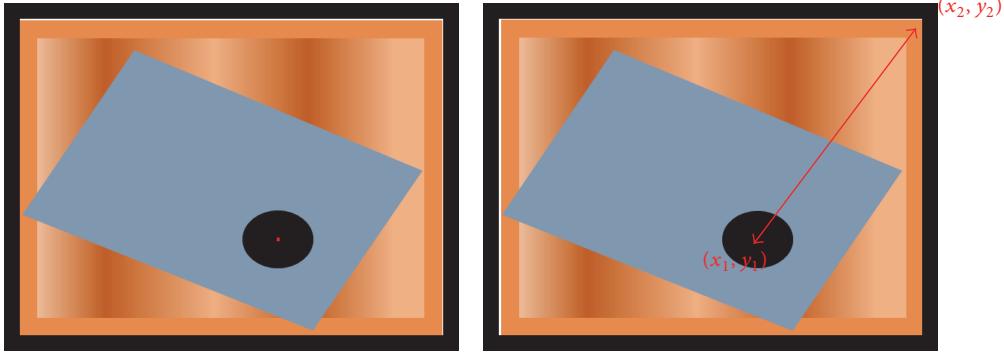


FIGURE 6: Distance to find in scaling.

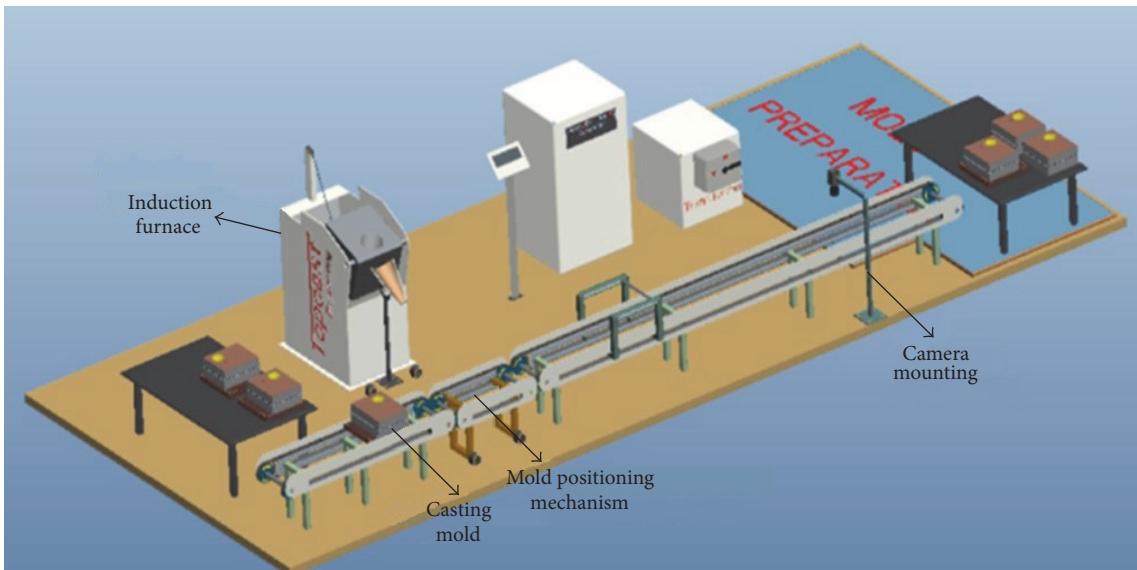


FIGURE 7: System model for automated sand casting.

pixels from the image and a relationship is made between millimeter and pixel:

$$\begin{aligned} n \text{ pixel} &= r \text{ mm} \\ 1 \text{ pixel} &= \frac{r}{n} \text{ mm.} \end{aligned} \quad (5)$$

Taking average of several iterations of the above practical approach gives an accurate result. After the distance in  $x$ - and  $y$ -direction between  $(x_1, y_1)$  and  $(x_2, y_2)$  has been found in  $s$ , the two values are sent to the external microcontroller based hardware through serial communication between the PC and the microcontroller. The microcontroller then uses these values to control the mechanical model in order to position the mold as desired.

### 3. Designing of Test Platform and Process Automation

**3.1. System Designing.** A complete system design mentioned in Figure 7, for the sand casting automation, was made in

Pro-Engineer software. The model is to the scale and is designed according to an induction furnace setup available in a manufacturing facility.

After successful model design an animation of the same is also made to visualize the concept of automation by mold positioning. The process starts by a mold preparation area, where casting molds are prepared; these are then transferred via a conveyor. During this first conveyor a camera takes the snapshot of the mold and processes it to find the location of pouring cup center. An alternate to the vision system is also shown in the model, in which a sensor scans the mold to find the pouring cup center, though this is not implemented here but is given just to show an alternate technique for future work. After conveyor 1 the second conveyor comes which can also move towards and back to the induction furnace. This conveyor is responsible for the actual mold positioning; a small scale model of it is also made to verify the concept and test the algorithm. After mold positioning and metal pouring this second conveyor moves the mold to the last conveyor which takes the mold to the cooling area.

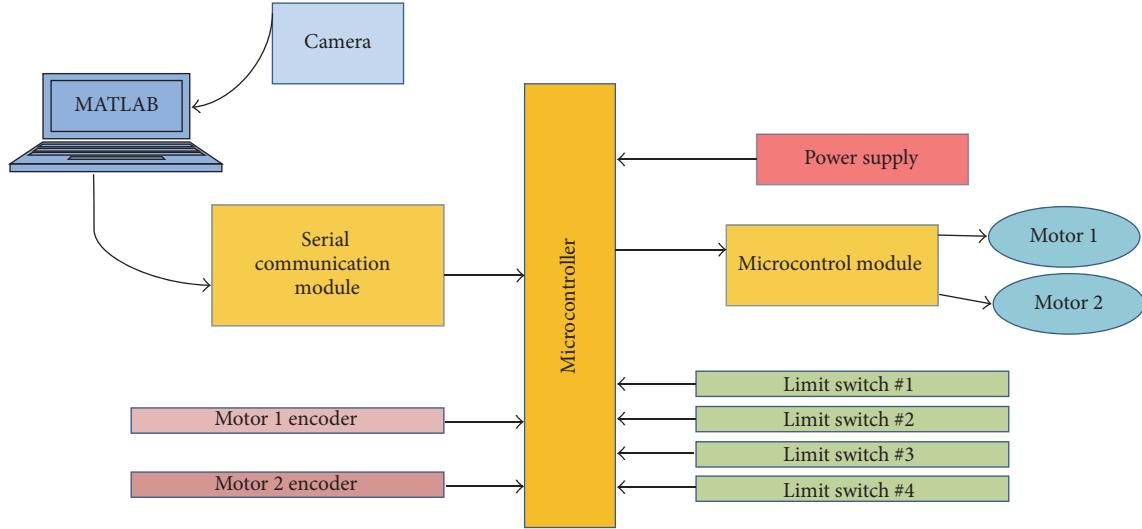


FIGURE 8: System electronic control with microcontroller nucleus linked with architecture.

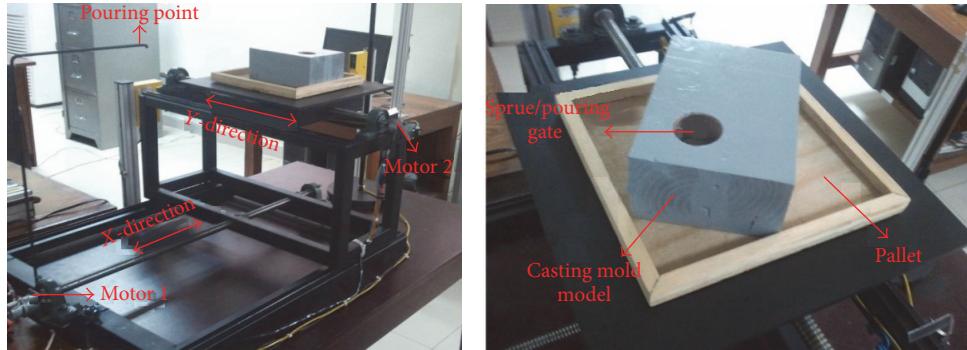


FIGURE 9: Mechanical model and positioning system.

**3.2. Scale Prototype Modeling.** The conveyor in the middle portion is the main part of the system which is responsible for carrying the mold towards the induction furnace and positioning it accordingly. Other two conveyors are just meant for transport; thus the small scale model (Figures 8, 9, 10, and 11) is made for them. The fabricated model consists of two lead screw mechanisms which give the mold place on top a two-degree-of-freedom positioning. The lead screws are run by DC-gear motors with encoder. Encoder provides pulses proportional to the rotation of lead screw and hence the linear distance traveled; hence they are used as a feedback for the microcontroller on the distance traveled by the mold as shown in Figure 9. Figure 8 gives a complete view of the system and it shows the linkage of different components.

The system consists of two portions; one is the vision system which is run on a computer, while the other is the microcontroller based hardware. To ensure successful operation of the system there is a serial communication link between these two portions. The output of the vision system is one of the inputs to the microcontroller hardware. The vision based system detects the position of the center of the mold pouring cup; this is basically an algorithm developed with a

camera connected to it. Picture is taken in real time and the algorithm processes it to find the center point. The second system is a microcontroller based embedded hardware; it takes the output of the vision system and transforms it into mechanical movement of the model. The microcontroller based embedded hardware is the communication module and a protocol between these two systems so that the data can be sent quickly and reliably.

**3.3. Mechanical Model and Positioning System.** The first step in the project was to develop the scale model of the system. Designing has been done by keeping in mind the actual equipment already present in industry and the dimensions of the actual area available for the setup of the new system. Initially there is an induction furnace, an induction heating generator, and a transformer present on the shop floor. Before starting modeling, the dimensions of the equipment were taken so as to produce a close to real life model. The software chosen for the modeling is Pro-Engineer v5. After modeling of the entire system to the scale, an animation of the same model has also been developed which shows the real time working of the model as an automated casting



FIGURE 10: Camera positioning mount above pallet.



FIGURE 11: Mold and pallet on mechanical model.

process. The animation has been developed in the animation module of PTC Pro-Engineer v5. The main components of the animation code are as follows:

- (i) Define initial condition of all mechanisms and moving parts.
- (ii) Define servo motors on all mechanisms and moving parts.
- (iii) Add servo motors on the timeline with defining start time, end time, speed, and initial position.
- (iv) Define body locks where required, and add them on time line with definition of start time, end time, lead body, and follower body.
- (v) Define viewing angles for each motion and add them over the entire timeline accordingly.

After successful model design, animation of the entire system was done. The next step was the designing and

fabrication of a small scale prototype model. The purpose of this prototype is to

- (i) verify the feasibility of the concept given in the model,
- (ii) check the working of the system in real time,
- (iii) get to know about the constraints and variables involved in the system,
- (iv) develop and test the algorithm which will run in the full scale system.

The main modules in the Prototype model are

- (1) mechanical model,
- (2) electronics hardware,
- (3) image processing algorithm.

As this is a small scale model, it is not possible to add an induction furnace to it; hence a fixed point is made with help of a rod which represent the pouring point of the induction furnace, and the molds thus have to position themselves such that the pouring cup center comes directly beneath this pouring point. Two side views of the system show the fabricated mechanical model (Figure 9).

After successful fabrication of the mechanical model, the electronics hardware needs to run with the main requirements as follows:

- (a) Microcontroller module.
- (b) Two motor control modules.
- (c) Module to allow communication between vision system and microcontroller.

For ease of understanding the practical system model is shown in Figures 10 and 11.

As shown in Figures 10 and 11, the mold which is placed on the top can be moved in X- and Y-direction for which two motors couples to the lead screws are present. The motors are mounted on a bracket which is fixed with the frame while the shaft on the motor fits directly in a hole made on the end of lead screw; the motor shaft is tightened with the help of a screw on the lead screw end. When the system would run the mold would be aligned automatically such that the center of the pouring cup would come beneath the pouring point as shown in Figure 11. A complete simulation of the system is also shown in Figure 12.

**3.4. Circuit Layout Designing.** Using all of the designing shown in Figure 13 the final layout of the printed circuit board was made. Lab Center Electronics Proteus has been used for the simulation, testing of the schematic, and the final designing of the PCB Layout. Firstly, the layout of each module has been designed individually and then they are merged together to form the final layout. Different color traces are used for ease of troubleshooting and understanding of Figure 13 is explained in Table 1.

The white connector on the left side is the DB9 connector which connects to the USB to serial converter, which is connected to the USB port of the PC. The wiring at the top

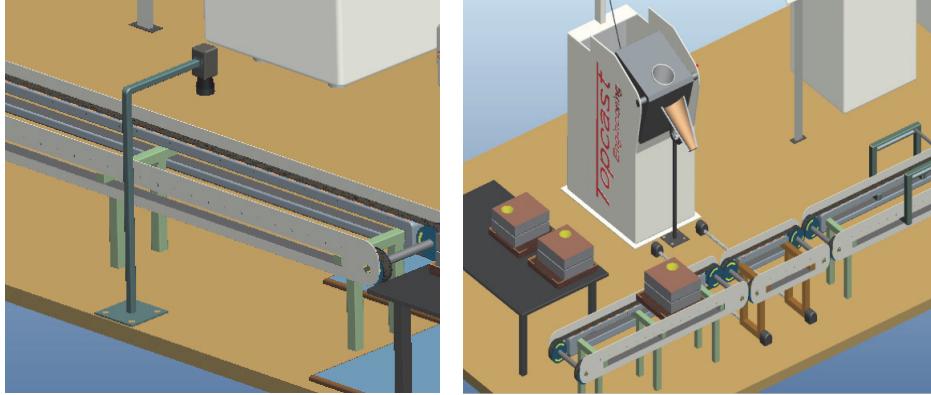


FIGURE 12: Simulated camera positioning on mold-pallet mechanical system.

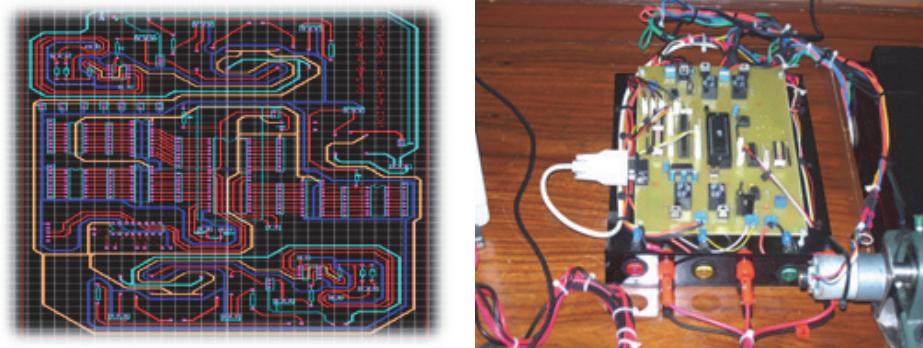


FIGURE 13: Proteus PCB layout fabricated electronics hardware.

TABLE 1: Color codes of layout.

Color	Function
Red	Data
Orange	Hardware power (5 V)
Blue	Ground of 5 V
Light green	Motor power (12–24 V)
Dark green	Ground for motor power

is going to the mechanical model; as shown in Figure 13 the entire wiring passes through the two sides of the housing; this makes the system neater and easier to troubleshoot if any problem occurs. The entire wiring of the circuit to the mechanical model has been done in a professional way, there is no soldering or hand joints in the entire wiring; wherever a joint is required between two or more wires, thimbles have been used and on top of those heat sleeves are inserted for insulation. Such wiring though takes time and patience but the outcome not only is a clean and tidy wiring job but also ensures the reliability of the system, as most of the problems occurring in such automation projects are due to poor wiring. Moreover as thimbles are used on every wiring joint, the system can be very easily dismantled

for transportation purposes and the same wiring harness can be used again to assemble it.

The final fabricated circuit having all the required modules and being wired up to the mechanical model is shown in Figure 13. After a lot of rechecking and verification of the layout with the original schematics has been done the layout was fabricated on a fiber PCB.

#### 4. Algorithm and Analysis

All the components of the prototype, mechanical model, electronics hardware, and vision system, were interfaced together as one system. The system was then fine-tuned by adjusting several parameters of the vision system and the microcontroller algorithm, to produce an automated mold positioning system. The flow of information in the entire system is worth mentioning; Figure 14 not only shows the flow of information but also shows all the modules and functions happening in the system in the correct sequence. The boxes in yellow show the functions executed in the vision system by MATLAB and in the green boxes are those executed by the microcontroller. After the start command is given in MATLAB till the actual physical mold positioning the process is fully automated without any human intervention.

After the distance values are in mm (millimeter) and sent to the microcontroller, it further modifies them into

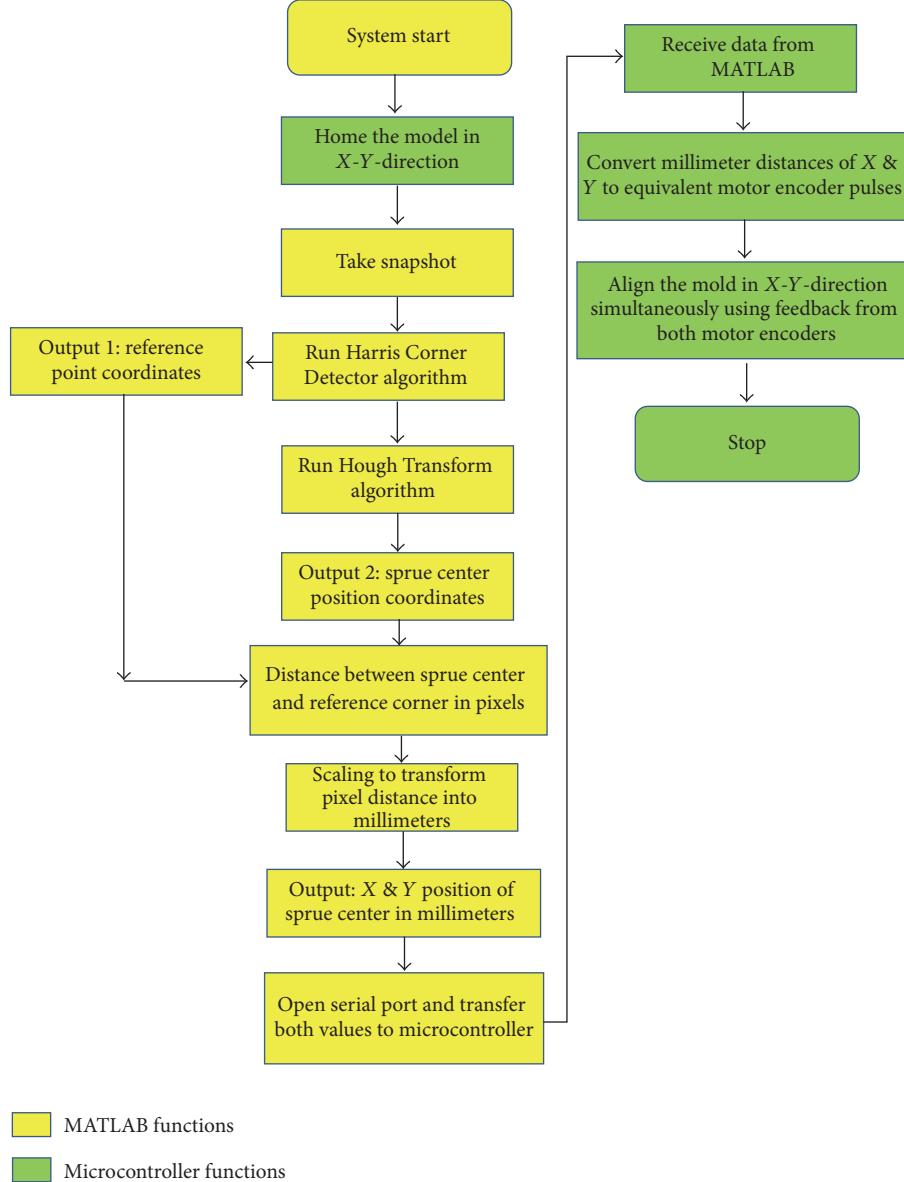


FIGURE 14: System flow of information.

coding that it can understand. The feedback from the two motors is in the form of pulses from the encoder, where the number of pulses is dependent on the number of rotations of the motor shaft. The pitch of the lead screw tells the linear distance traveled in one rotation of the lead screw; thus a relationship is formed between the encoder pulses and the linear distance to travel. The values thus sent by MATLAB are converted to equivalent motor encoder pulses, which the microcontroller monitors to achieve the distance desired. The system developed here is unique and better than other sand casting automation setups as it can accommodate the following variations in the molds without any human interventions or hardware changes to mechanism shown in Figure 6 and summarized in Figure 12 in form of flow graph:

- Different mold position on the pallet
- Different mold orientation on the pallet
- Molds of different sizes
- Molds with different pouring cup positions
- Molds with different pouring cup radius.

In a sand casting setup where wide variety of parts are made on the same assembly line, it requires the molds to be of different sizes, pouring cup location, or radius to vary; thus this system comes into play in such situations, and it autonomously handles the variations in the molds. Moreover, the worker who is preparing the molds and putting them on the conveyor belt does not have to think or put any effort on

the placement of the molds; the system will accept molds in any orientation or position on the pallet

## 5. Discussion and Conclusion

In previous sand casting automation research studies, implementation of precise pouring of the molten metal requires molds on the conveyor line to be same size, having same pouring cup location and radius. But if molds have to be casted which are of different sizes, the entire production line has to be stopped, and after physical adjustment to the pouring mechanism the system is started to cater for this limitation. Our present machine vision based sand casting automation system does not require any sort of production stopping or manual adjustment once it is started; it will keep the casting process running regardless of the variations in the geometry and position of the casting mold.

In this research project, first a complete model of the automation system is formulated by which this new scheme of automation can be implemented in the sand casting industry. Then as a proof of concept, to test the mechanical feasibility and test and verify the algorithms developed, a small scale prototype model is made on which successful demonstration is given on automatic mold positioning to be used in a complete sand casting system model with machine vision methodology. The automatic mold positioning prototype has proved to be an achievement due to its sharp and reliable mold positioning capability. The system presented here can be coupled with already presented automatic metal pouring systems in the industry, thus giving a fully automatic sand casting system. The introduction of mold positioning system presented here will increase the flexibility, safety, and mold handling capability of existing automated sand casting setups, thus allowing different parts to be casted on the same conveyor line without any human intervention or process stopping.

The system presented here is not limited just to a sand casting process; other industries requiring an autonomous positioning system for their parts/items on a conveyor system can also benefit from this research.

## Competing Interests

The authors declare that they have no competing interests.

## References

- [1] H. Kosler, U. Pavlovčić, M. Jezeršek, and J. Možina, “Adaptive robotic deburring of die-cast parts with position and orientation measurements using a 3D laser-triangulation sensor,” *SV-Journal of Mechanical Engineering*, vol. 4, pp. 207–212, 2016.
- [2] E. Neuman and D. Trauzeddel, “Pouring systems for ferrous applications,” *Foundry Trade Journal*, pp. 23–24, 2002.
- [3] A. Ito, Y. Noda, and K. Terashima, “High-precision liquid pouring control while keeping lower ladle position and avoiding clash with mold,” in *Proceedings of the IEEE International Conference on Control Applications (CCA ’12)*, pp. 246–251, Dubrovnik, Croatia, October 2012.
- [4] X. Zhu, R. Chen, and Y. Zhang, “Automatic defect detection in spring clamp production via machine vision,” *Abstract and Applied Analysis*, vol. 2014, Article ID 164726, 9 pages, 2014.
- [5] L.-H. Zou, C. Jie, Z. Juan, and L.-H. Dou, “The comparison of two typical corner detection algorithms,” in *Proceedings of the 2nd International Symposium on Intelligent Information Technology Application (IITA ’08)*, vol. 2, pp. 211–215, Shanghai, China, December 2008.
- [6] Y. Choe, H.-C. Lee, Y.-J. Kim, D.-H. Hong, S.-S. Park, and M.-T. Lim, “Vision-based estimation of bolt-hole location using circular hough transform,” in *Proceedings of the ICROS-SICE International Joint Conference*, pp. 4821–4826, Fukuoka International Congress Center, Fukuoka, Japan, August 2009.
- [7] Z. Zhifeng, “Measuring diameter of non-threaded hex bolts based on hough transform,” in *Proceedings of the 3rd International Conference on Measuring Technology and Mechatronics Automation (ICMTMA ’11)*, pp. 526–528, Shanghai, China, 2011.
- [8] U. Asgher, R. Ahmad, and S. I. Butt, “Mathematical modeling of manufacturing process plan, optimization analysis with stochastic and DSM modeling techniques,” *Advanced Materials Research*, vol. 816–817, pp. 1174–1180, 2013.
- [9] Y. Noda, K. Terashima, M. Suzuki, and H. Makino, “Weight control of pouring liquid by automatic pouring robot,” in *Proceedings of the IFAC Workshop on Automation in Mining, Mineral and Metal Processing*, vol. 70, pp. 1–6, Vina del Mar, Chile, 2009.
- [10] Y. Noda and K. Terashima, “Modeling and feedforward flow rate control of automatic pouring system with real ladle,” *Journal of Robotics and Mechatronics*, vol. 19, no. 2, pp. 205–211, 2007.
- [11] A. Wilson, Ed., *System Checks Plastic Molding Consistency*, 2007, <http://www.vision-systems.com/articles/print/volume-12/issue-2/features/profile-in-industry-solutions/system-checks-plastic-molding-consistency.html>.
- [12] U. Asgher, H. Muhammad, M. M. Hamza, R. Ahmad, S. I. Butt, and M. Jamil, “Robust hybrid normalized convolution and forward error correction in image reconstruction,” in *Proceedings of the 10th International Conference on Innovations in Information Technology (IIT ’14)*, pp. 54–59, Al-Ain, UAE, November 2014.
- [13] J. Chi, L. Liu, J. Liu, Z. Jiang, and G. Zhang, “Machine vision based automatic detection method of indicating values of a pointer gauge,” *Mathematical Problems in Engineering*, vol. 2015, Article ID 283629, 19 pages, 2015.
- [14] H. M. Yip, J. C. Li, K. Xie et al., “Automated long-term monitoring of parallel microfluidic operations applying a machine vision-assisted positioning method,” *The Scientific World Journal*, vol. 2014, Article ID 608184, 14 pages, 2014.
- [15] S. Srivastava, S. Boyat, and S. Sadistap, “A robust machine vision algorithm development for quality parameters extraction of circular biscuits and cookies digital images,” *Journal of Food Processing*, vol. 2014, Article ID 376360, 13 pages, 2014.



Journal of  
Nanotechnology



International Journal of  
Corrosion



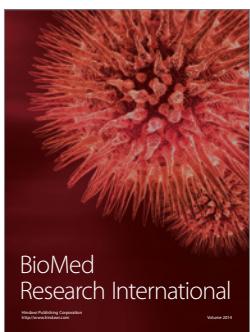
International Journal of  
Polymer Science



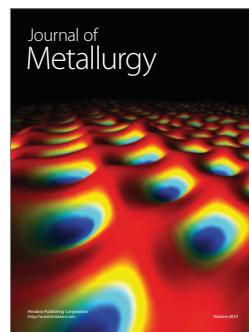
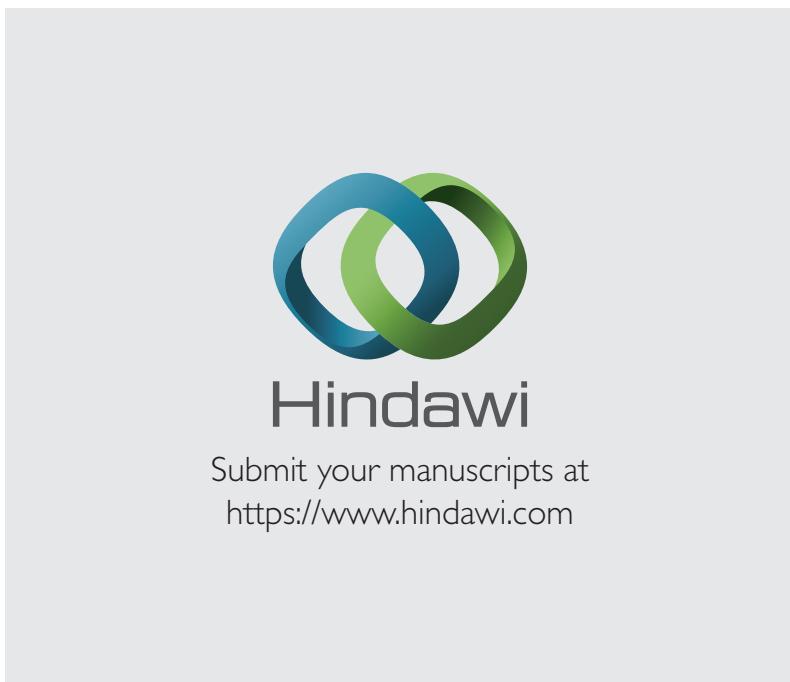
Smart Materials  
Research



Journal of  
Composites



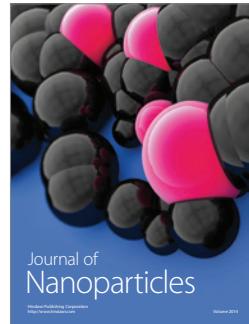
BioMed  
Research International



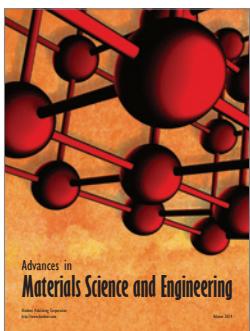
Journal of  
Metallurgy



Journal of  
Materials



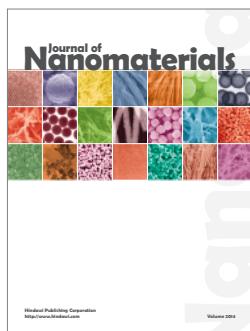
Journal of  
Nanoparticles



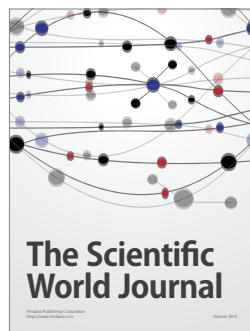
Advances in  
Materials Science and Engineering



Scientifica



Journal of  
Nanomaterials



The Scientific  
World Journal



International Journal of  
Biomaterials



Journal of  
Nanoscience



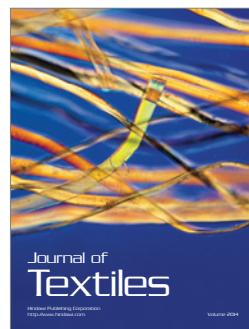
Journal of  
Coatings



Journal of  
Crystallography



Journal of  
Ceramics



Journal of  
Textiles