

Remote Retail Monitoring and Stock Assessment using Mobile Robots

Swagat Kumar, Geetika Sharma, Nishant Kejriwal,
Saumil Jain, Madhvi Kamra, Brijendra Singh and Vishal Kumar Chauhan
Innovation Lab, Tata Consultancy Services, New Delhi, India

Email: {swagat.kumar, geetika.s, nishant.kejriwal, saumil.jain, madhvi.kamra, vishalkumar.chauhan}@tcs.com

Abstract—This paper describes a Virtual Reality (VR) based system for automating data collection and surveying in a retail store using mobile robots. The manpower cost for surveying and monitoring the shelves in retail stores are high, because of which these activities are not repeated frequently causing reduced customer satisfaction and loss of revenue. Further, the accuracy of data collected may be improved by avoiding human related factors. We use a mobile robot platform with on-board cameras to monitor the shelves either autonomously or through tele-operation. A remote operator can control the robot from a console which shows a 3D view of the store as well as, capture real images and videos of the store. The robot is designed to facilitate automatic detection of Out-of-Stock (OOS) situations. It would be possible for a single operator to control multiple robots placed at different stores thus optimizing the available resources. As the deployment of the proposed system does not require modifying existing infrastructure of the store, the cost of the entire solution is cheaper with shorter return-on-investment (ROI) period.

Index Terms—Retail Robotics, Service Robots, Remote Operation, Mobile Robots, Out-of-Stock (OOS)

I. INTRODUCTION

Temporary out-of-stock (OOS) situation is considered to be an important problem in the retail industry. It is estimated that the global average out-of-stock rate is about 8% and it costs retailers about 4% losses in sales [1] [2]. The out-of-stock situations may arise because of several reasons. However, it is estimated that about 70-90% of stockouts are caused by defective shelf replenishment practices as opposed to 10-30% resulting from the problems in the supply chain [1]. The former case leads to shelf-OOS while the later leads to store-OOS [3]. We are primarily concerned with the case when the items are available in the warehouse or the backroom inventory of the store but they are out-of-stock on the shelves. This is more frequent with fast moving consumer goods (FMCG) which are depleted faster than their replenishment. Higher frequency of checking can reduce the OOS related problems to a greater extent. Currently, most of these surveys are carried out by humans at pre-defined intervals. Hence, a higher frequency of surveys would lead to higher cost leading to lesser profit margins. Moreover, the data collected through humans is erroneous and unreliable. RFID based technologies have been found to be useful in dealing with OOS situation apart from automating the supply chain management [4] [5]. Some of the companies like NeWave [6] and Shelfx [7] provide smart shelf solutions based on RFID that can detect OOS in real-time apart from facilitating automatic checkout.

However, use of RFID based technologies require altering the store environment to accommodate antennae, sensors etc. and thus require more time and money for deployment. Moreover, RFID-based solutions which require item-level tagging is still expensive for low cost grocery items.

In this context, we propose to use mobile robots to carry out these surveys and detect shelf-OOS in real-time as well as on-demand. Similar attempt has been made by Priya Narasimhan's group in their AndyVision project [8]. To some extent, we have been inspired by their work. However, our focus is on providing a robust solution by keeping a human in the loop. The robots are partially or completely controlled by a remote operator. Our solution is not only cost-effective but also, does not require any modification to the existing infrastructure. Apart from detecting shelf-OOS and misplaced items, a mobile robot can provide several other value added services like providing useful information to customers in the form of promotions or discounts. It may also be used as a platform to facilitate automatic check-out thereby avoiding queues at point of sales (POS) counters. It could also be used for warehouse monitoring and surveillance during night time. Some of the benefits of using robots in retail store have been described in [9].

The claims or novel contributions made in this paper are as follows: (1) The idea of monitoring the shelves using remotely operated robots is a new concept. (2) We rely primarily on on-board sensors for data collection thereby minimizing the changes needed in the environment it has to operate in. (3) We create 3D/2D virtual environment which is updated to reflect the current robot position in the real world, thereby reducing the cognitive load, of the moving robot, on the operator. The presence of a human operator in the loop provides robustness to the entire solution compared to a robot-only solution without any significant increase in the cost.

In this paper, we describe the implementation of a Proof-of-Concept carried out in our lab. The usability is demonstrated through various experiments. The use of low cost hardware makes the whole solution affordable and thus, viable as a commercial solution for retail store monitoring. The rest of this paper is organized as follows. An overview of related literature is provided in the next section. The main idea and the proof-of-concept is described in Section III. The details of hardware and software systems are provided in Section IV and V respectively. The experimental results are provided in Section VI followed by conclusion in Section VII.

II. RELATED WORKS

This section provides an overview of various related works where robots have been used in the retail environment. Kamei *et. al.* [10] [11] [12] use a network robot system to incorporate recommendation methods used in e-commerce system into a real-world retail shop. While sensors like LRF and cameras are used to analyze pre-purchase behaviour of customers, the physical robots are used to present recommendations and show directions. In another work, Matsuhira *et. al.* [13] develop a robotic transport system to assist people during shopping. The system consists of two robots - one of them follows the person while the other acts as shopping cart carrying purchased items. The on-board robot sensors and environmental cameras are used to facilitate human-robot interaction. Similarly, RoboCart [14] assists visually impaired customers navigate a typical grocery store and carry purchased items. It relies on RFID tags for localization and laser for navigation.

There are few related patents as well. Bancroft *et. al.* [15] developed a mobile robot system for interacting with customers and assist them by providing useful informations. The mobile robot was capable of travelling from one location to another and could accept inputs from a customer and give output verbally or in written messages. Ostrowski [16] and Hudnut *et. al.* [17] developed a shopping cart which could detect the items lying on the bottom of the shopping cart. It used a visual sensor identify products using visual features like SIFT. There objective was to speed up the check out process. Konstad *et. al.* [18] developed a similar shopping cart system which could detect object using RFID and weight sensors. The US patent by Razumov [19] describes a novel robotic system for selling goods at a retail facility having a storage area not accessible by customers. The system includes multiple robots and a control system. Based on the order placed by a customer, the control system assigns at least one robot to each customer which picks up multiple items from separate places and brings it to the customer. We are excluding the robotic inventory systems as in [20] [21] [22]. Apart from AndyVision [9], none of the above robots can be used for carrying out surveys within a retail store. AndyVision can precisely localize itself using external sensors. It can update the planogram in real-time. It can detect empty shelves or misplaced items, recognize labels and products. The computationally intensive tasks are off-loaded to a cloud server. AndyVision project aims to revolutionalize the retail stores by using robots to automate all aspects of retail store management. However, the commercial viability of such a system is still questionable at this point.

Our primary focus is to provide a commercially viable solution to the problem of retail stock monitoring. While it is always possible to build a sophisticated robot to do everything on its own, presence of a human operator in the loop can not only make the solution robust against various unseen circumstances, it can reduce the overall cost of the solution significantly and thus, making it acceptable to the market. The details of the proposed scheme is discussed next in this paper.

III. THE IDEA

We propose to use a mobile robot to automate the data collection in a retail store environment. We plan to use on-board cameras to collect images and videos of the shelves which are transmitted to remote server over internet or LAN. These images will be processed over cloud to generate several analytics. The use of robots for retail monitoring is expected to provide following benefits: (1) It will be possible to carry out surveys more frequently thereby reducing the cost per survey. (2) It will increase the reliability and authenticity of data being gathered. (3) A robot can provide other value added services which will enrich the shopping experience of customers.

The schematic of a retail robotics framework is shown in Figure 1. The robots can be controlled and managed by a remote call centre providing round-the-clock service. The product manufacturers can obtain survey reports from these call centres as and when required. A customer can obtain various product related information from the call centre through telephone calls or from servers using mobile apps. The user can also get the real-time data availability of a product from the nearest store through this system. In this paper, we only discuss the implementation of a part of the solution.

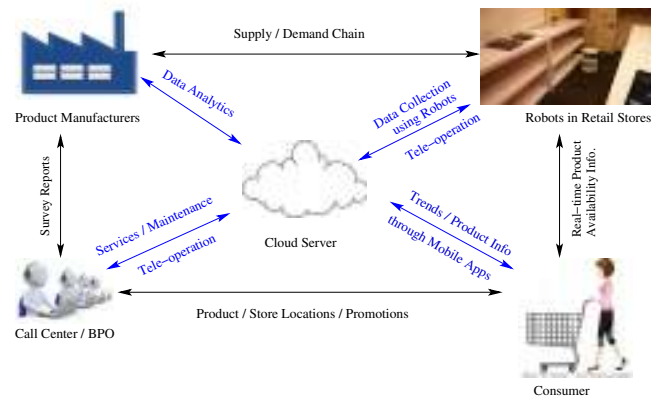


Fig. 1. A cloud-based retail robotics framework. Robots are used for collecting data from retail stores in real-time which can be used for improving the efficiency of the entire retail ecosystem.

As a proof-of-concept, we set up a mock store where a tele-operated robot is used for gathering visual data. The robot can operate in a tele-operation mode as well as goal-based autonomous navigation mode. Through the operator console, the remote operator can obtain live stream of videos and images recorded through on-board camera. The operator can capture images of a shelf whenever required. The operator has a virtual 3D environment where the current robot positions are continuously updated where the robot moves in the real environment. The virtual 3D environment is created using planogram information and images obtained from the remote store. The store model is created once, but the planogram can change depending on products being displayed. The operator console has arrow keys for robot navigation. It also has pre-defined goals that can be given to the robot for autonomous

navigation. The Out-of-Stock (OOS) and misplaced-items are detected by processing the images over a remote server. Robot locations and Shelf status is also made available through a 2D grid-map. A 2D map is useful in those cases where limited computational resources are available at the operator's location.

IV. THE SYSTEM HARDWARE

In order to demonstrate the proof-of-concept, we set up a mock retail store in our lab. The robotic system consists of a Turtlebot [23] having an on-board Kinect camera for navigation and a low-power netbook for running essential algorithms. It uses two USB cameras to capture images and videos of the racks on either side of the robot. The Turtlebot with cameras are shown in Figure 2. The robot is controlled through a operator console (described later) running on a remote computer. The remote computer running the operator console communicates with the robot using wireless network. Surveillance cameras available in the store could be used for obtaining the bird's eye view of the store. It could be used for improving the localization of robot as has been done in [9] [14].



Fig. 2. A Turtlebot with a pair of on-board USB cameras (shown in red circles) used for monitoring racks.

V. THE SOFTWARE SYSTEM

A. The Software Architecture

The block diagram of the entire application is shown in Figure 3. It has two parts - one physical system consisting of a real robot and, a second part comprising of a virtual environment with graphical user interface. These two systems communicate over a wireless network. The software architecture of this application consists of three basic components: a server, a client and a communication network. The software modules are developed in C/C++ using ROS [24]. ROS is an open-source distributed software framework which provides standard operating system services for a robot. Any robot system generally comprises of many nodes where each node is a process performing a particular task. All the nodes communicate with each other and with the ROS Master by passing messages. The ROS Master provides lookup information about other nodes connected with it. ROS is based on TCPROS network communication protocol which uses standard TCP/IP sockets. In this application, robot in the retail store acts as a server and a remote machine acts as a client. Virtual environment of the retail store and operator console runs on the remote client machine.

The ROS graph showing all the nodes and connections is shown in Figure 4. The application has three important modules which comprises of several nodes running either on

the server or on one of the many clients. The first module (labeled as '1') is the navigation module which is used for autonomous navigation of the robot in real world. It uses kinect to generate a fake laser data. This laser data is used along with the map obtained from `map_server` node for localization. The static map of the store is generated using ROS `gmapping` stack. The second module (labelled as '2') updates the robot position in the virtual 3D environment created using Gazebo. We implement Dijkstra algorithm to find path between the current node (cell) to the goal node. It uses the odometry obtained from on-board local algorithm (`amcl`) to find the nearest cell in a 2D grid. The third module (labelled as '3') runs the graphical user interface (GUI) through which an user can control the actual robot. It runs as a node `talker` which provides goals to the robot through `simple_navigation_goals` node. This module is also used for capturing live images and videos on-demand.

B. Graphical User Interface

The graphical user interface is used by an operator to control or monitor a robot located at a remote retail store. The operator console appears as shown in Figure 5. The operator can manually control the robot using arrow keys. It is also possible to provide pre-defined goal locations for autonomous robot navigation. Apart from viewing the views transmitted from on-board cameras, the updated robot position is available from the 3D virtual environment.

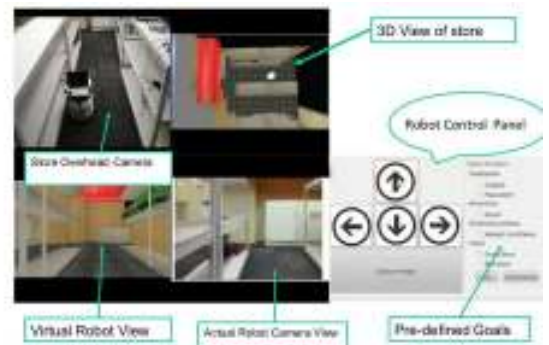


Fig. 5. Remote Operator Console

C. Creating a 3D virtual environment

In this section we describe our technique for creating a 3D model of the retail store. The store model needs to (a) mirror the real store in terms of its structural and interior layout and (b) have items on virtual shelves placed according to the stores planogram. Both of these requirements are to aid the remote operator's tasks of controlling the robot and determining the whether items have been placed on shelves correctly.

We use a floor plan of the store, as shown in figure 6 (a), to compute its structural layout. We assume that walls are represented as thick, straight lines in the plan and symbols or geometric shapes are used to represent other objects such as racks. These are detected in the floor plan image using simple



Fig. 7. (a) 3D model of store with textures (b) With objects placed on shelves

D. Synchronization between real and virtual world

Gazebo [25] is utilized to create a 3D virtual environment for the retail store. The simulated robot and the actual robot could be driven with the same velocity command provided through `cmd_vel` node. However, the same velocity leads to a motion in real world which is quite different from that in the virtual world. This is due to the fact that the physical properties of the virtual environment might differ significantly from those of the actual world. In order to synchronize the motion of the robot between the real and the virtual world, we create two separate velocity nodes in the ROS graph. One drives the actual robot and the other drives the virtual robot. The motion commands for the virtual robot is generated using Dijkstra path planning algorithm [26]. The virtual 2D space is divided into equally spaced grids. The latest robot coordinate is obtained from the corrected odometry which is available from the localization algorithm running on the actual robot. The shortest path between robot's current location in the virtual world and the goal position obtained from real robot is obtained using Dijkstra's algorithm.

VI. EXPERIMENTAL RESULTS

A. The experimental Setup

The experiment is carried out in a mock retail setup. The room has a size of $4.8 \text{ m} \times 3.4 \text{ m} \times 2 \text{ m}$. It has six racks ($1.5 \text{ m} \times 0.5 \text{ m} \times 0.9 \text{ m}$) forming three rows. In this case, the robot has one closed-loop path in the retail store. The robot used for the experiment is a Turtlebot [23] which carries two USB cameras facing the racks on either side of the robot path. The total cost of the robot is about US\$ 1500 and the whole set up may not cost more than US\$ 2000. A simple calculation can show that the return on investment could be obtained in less than a year, if the robotic solution can reduce the manpower requirement even by a small number. The two on-board USB cameras could be used for monitoring both the shelves lying on either side of the robot simultaneously. The views from both the cameras are transmitted to the operator's console and they appear as shown in Figure 8. This scheme makes it easier and faster to monitor shelves in a retail store. The operation of the entire application is demonstrated through videos made available through YouTube [27] [28]. The first video [27] shows various views available at the operator's console. It shows the robot's location in virtual as well as the actual environment. The robot's camera view along with

the virtual view is also shown. The operator has the option to operate it manually or provide a pre-defined goal for the robot to navigate autonomously. The second video [28] shows how a tele-operated robot could be used for visual inspection of racks.



Fig. 8. Two instances of rack images obtained from on-board USB cameras.

B. Localization and Mapping

The map of the environment is created using the ROS `gmapping` algorithm. The depth map obtained from Kinect is used as a fake laser scan for generating the map. Once the map is available, current laser scan is used for localizing the robot. The robot position obtained from the corrected odometry (through localization) for four subsequent laps of traversal is shown in Figure 9. The average error in robot position is approximately 20cm over four laps. This error could be further reduced by using external sensors like RFID or wireless antennae as done in [9] [14]. External surveillance cameras present in stores could also be used for improving the accuracy of robot localization.

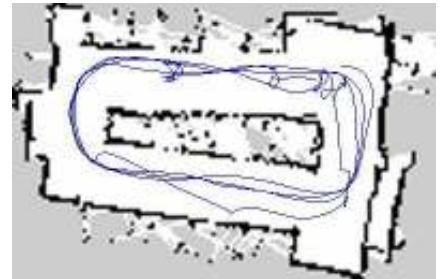


Fig. 9. Robot path during multiple laps of traversal obtained from the corrected odometry. Black pixels show the edges of obstacle obtained from the fake laser creating using Kinect depth readings.

C. Updating robot position in 2D Grid map

As discussed above, the robot position is updated regularly in the virtual environment available at the operator's console. Rendering and updating information in a 3D environment requires more computational resources. While a 3D environment provides visually rich look and feel of a real environment, it might not be needed if we are only concerned about robot's location in a 2D map. Keeping this in mind, we also have a 2D grid map where the robot position is updated and synchronized with the actual robot position. The available area is divided

into grids. The grids which are partially occupied and are smaller than the dimension of robot are considered as non-traversable as shown in Figure 10. The robot is shown as a small black circle. The corrected odometry obtained from SLAM algorithm is used to find the closest grid.

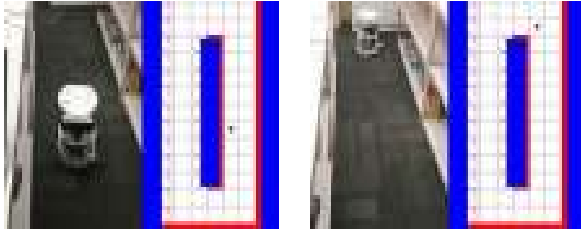


Fig. 10. The actual robot position is updated on a 2D grid map. Blue colour represents obstacles corresponding to the racks. The red colour shows the non-traversal area for the grids which have more than 50% occupancy. Circles with crosses show the grids which have less than 50% occupancy but are smaller than the robot size and hence non-traversable.

VII. CONCLUSION

In this paper, we provide an implementation of a robotic system which can be used for carrying out surveys and stock assessment in retail stores with reduced manpower. The entire solution is implemented with a low cost robot costing approximately US\$2000. The robot could be tele-operated by a remote operator to monitor and check individual shelves without moving away from his desk. He can also control multiple robots from the same console. The robot can also be operated in autonomous mode taking pictures and videos which could be processed over a cloud to generate statistics. Currently, our focus is on detecting OOS situation or misplaced items on the racks and generate alarms in real-time. The remote operator has a virtual 3D/2D environment where the robot location is updated in real-time to show its current location. The operator also has flexibility of viewing and storing videos as seen by the on-board cameras. This human-in-loop solution provides robustness to the whole solution making it commercially viable for deployment in real-world scenarios.

REFERENCES

- [1] Wikipedia, "Stockout or out-of-stock," <http://en.wikipedia.org/wiki/Stockout>.
- [2] H. Che, J. Chen, and Y. Chen, "Investigating effects of out-of-stock on consumer sku choice," 2011.
- [3] T. W. Gruen and D. S. Corsten, *A comprehensive guide to retail out-of-stock reduction in the fast-moving consumer goods industry*. Grocery Manufacturers of America, 2007.
- [4] M. Kärkkäinen, "Increasing efficiency in the supply chain for short shelf life goods using rfid tagging," *International Journal of Retail & Distribution Management*, vol. 31, no. 10, pp. 529–536, 2003.
- [5] D. Corsten and T. Gruen, "Desperately seeking shelf availability: an examination of the extent, the causes, and the efforts to address retail out-of-stocks," *International Journal of Retail & Distribution Management*, vol. 31, no. 12, pp. 605–617, 2003.
- [6] NeWave, "RFID-based smart shelf system," <http://newavesensors.com/>.
- [7] ShelfX, "Smart shelf system," <http://www.shelfx.com/product/retail>.
- [8] K. Mankodiya, R. Gandhi, and P. Narasimhan, "Challenges and opportunities for embedded computing in retail environments," in *Sensor Systems and Software*. Springer, 2012, pp. 121–136.
- [9] K. Mankodiya, R. Martins, J. Francis, E. Garduno, R. Gandhi, and P. Narasimhan, "Interactive shopping experience through immersive store environments," in *Design, User Experience, and Usability. User Experience in Novel Technological Environments*. Springer, 2013, pp. 372–382.
- [10] K. Kamei, K. Shinozawa, T. Ikeda, A. Utsumi, T. Miyashita, and N. Hagita, "Recommendation from robots in a real-world retail shop," in *International Conference on Multimodal Interfaces and the Workshop on Machine Learning for Multimodal Interaction*, ser. ICMI-MLMI '10. New York, NY, USA: ACM, 2010, pp. 19:1–19:8.
- [11] K. Kamei, T. Ikeda, H. Kidokoro, M. Shiomi, A. Utsumi, K. Shinozawa, T. Miyashita, and N. Hagita, "Effectiveness of cooperative customer navigation from robots around a retail shop," in *Privacy, security, risk and trust (passat), 2011 IEEE third international conference on and 2011 IEEE third international conference on social computing (socialcom)*. IEEE, 2011, pp. 235–241.
- [12] K. Kamei, T. Ikeda, M. Shiomi, H. Kidokoro, A. Utsumi, K. Shinozawa, T. Miyashita, and N. Hagita, "Cooperative customer navigation between robots outside and inside a retail shop: An implementation on the ubiquitous market platform," *annals of telecommunications-Annales des télécommunications*, vol. 67, no. 7-8, pp. 329–340, 2012.
- [13] N. Matsuhira, F. Ozaki, S. Tokura, T. Sonoura, T. Tasaki, H. Ogawa, M. Sano, A. Numata, N. Hashimoto, and K. Komoriya, "Development of robotic transportation system - shopping support system collaborating with environmental cameras and mobile robots," in *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK)*. IEEE, 2010, pp. 1–6.
- [14] V. Kulyukin, C. Gharpure, and J. Nicholson, "Robocart: toward robot-assisted navigation of grocery stores by the visually impaired," in *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, 2005, pp. 2845–2850.
- [15] A. Bancroft and C. Ward, "Methods for facilitating a retail environment," Apr. 17 2007, US Patent 7,206,753. [Online]. Available: <https://www.google.com/patents/US7206753>
- [16] J. Ostrowski, L. Goncalves, M. Cremean, A. Simonini, A. Hudnut, *et al.*, "System and methods for merchandise checkout," Sept. 5 2006, US Patent 7,100,824.
- [17] A. Hudnut, A. Simonini, M. Cremean, H. Morgan, *et al.*, "Method of merchandising for checkout lanes," July 24 2007, uS Patent 7,246,745.
- [18] R. A. Konstad and J. W. Lawrence, "Shopping cart," July 10 2007, US Patent 7,242,300.
- [19] S. Razumov, "Robotic retail facility," Oct. 27 2005, US Patent App. 10/832,383. [Online]. Available: <https://www.google.com/patents/US20050238465>
- [20] W. DAVIDSON, "Rail-mounted robotic inventory system," Sept. 19 2013, WO Patent App. PCT/US2013/029,958. [Online]. Available: <https://www.google.com/patents/WO2013138193A2?cl=en>
- [21] B. Adelberg, W. Smith, J. Hutton, S. Viswanathan, and S. Lusardi, "Vending store inventory management and reporting system," Apr. 29 2010, WO Patent App. PCT/US2009/061,623. [Online]. Available: <https://www.google.com/patents/WO2010048375A1?cl=en>
- [22] R. D'Andrea, P. Mansfield, M. Mountz, D. Polic, and P. Dingle, "Method and system for transporting inventory items," Oct. 2 2012, US Patent 8,280,547. [Online]. Available: <https://www.google.com/patents/US8280547>
- [23] Turtlebot, "Open-source robot development kit for apps on wheels," <http://www.turtlebot.com>.
- [24] ROS, "Robot operating system," <http://www.ros.org>.
- [25] Gazebo, "A multi-robot simulator," <http://www.gazebosim.org/>.
- [26] Wikipedia, "Dijkstra's algorithm," http://en.wikipedia.org/wiki/Dijkstra%27s_algorithm.
- [27] N. Kejriwal, "Youtube video on retail monitoring using tele-operated robot," <https://www.youtube.com/watch?v=21wDfvOm20Q>.
- [28] —, "Youtube video on rack inspection using a tele-operated robot," http://www.youtube.com/watch?v=Zflx2kg8P_k.