Detection and Avoidance of Pedestrians

The Need for Pedestrian Safety: Facts and Figures

In the United States in 2006 4,784 pedestrians were killed in traffic accidents. Because a larger percentage of Americans use private vehicles than walk or use public transportation compared to people in many countries in the European Union and around the world, less attention is given in the U.S. to pedestrian safety than in these more pedestrian-friendly countries. In addition, though numerous efforts have been made to improve vehicle passenger safety (air bags, crumple zones, et cetera), no similar effort to protect pedestrians has been initiated. While pedestrians were involved in just over 1% of all traffic accidents in 2006, these vulnerable road users accounted for over 12% of all traffic fatalities.

Most of the accidents in which pedestrians were involved took place at night in urban areas under normal weather conditions. In almost half of all fatal pedestrian crashes alcohol was involved, usually on the part of the pedestrian. [1]

Motor vehicle collisions are the leading cause of fatal accidents and one of the leading causes of non-fatal (but still damage or injury causing) accidents in the Army. Although the Army is making a decided effort to decrease the number of fatalities from traffic mishaps 65% of fatalities were still from driving accidents in 2006. Under the assumption that the same percentage of these accidents involves pedestrians as in civilian statistics, increased pedestrian safety would greatly benefit the Army and save the lives of many soldiers. [2]

A much larger percentage of traffic accidents in Europe and other nations involve pedestrians than in the United States. This is unsurprising given the greater percentage of pedestrians in many other countries, especially low income countries, compared to most of the US. The World Bank reports that globally, more than 0.76 million pedestrians are killed in traffic accidents each year, 35% of which are children. [3]

These statistics show the dire need to increase pedestrian safety but also point the specific scenarios (e.g. at night in urban areas) where targeted pedestrian safety improvements would benefit the most people.

Project Summary

The purpose of this project is to design and implement a pedestrian detection and avoidance system for use in civilian and military applications. The proposed system will function in both structured environments, such as road lanes, and unstructured environments, such as fields and empty parking lots. Both of these environments are encountered by civilians and military personnel and a system versatile enough for both functions would be preferable to one with a more limited range of use. Due to the statistics mentioned above, the system will be optimized for nighttime driving in urban areas under normal weather conditions, as this situation is the most dangerous for pedestrians.

Based on the literature surveyed, a sensor suite consisting of a far infrared camera, a visible light (CCD) camera and a scanning laser range finder appear to be the most reliable and accurate choice of sensors for the cost. Sensor fusion performed between these devices will lead to robust detection and tracking under most weather conditions.

Pedestrians, stationary and dynamic obstacles, and road markings will be detected by the sensor suite and tracked via a Kalman filter. The system will respond to the presence of obstacles by appropriately altering the vehicle's velocity in a structured environment and planning a path to avoid obstacles when in an unstructured environment.

Previous Comprehensive Projects

From 2000 until 2005, the European Union partnered with industry to fund vulnerable road user (VRU) protection projects. The first project, Preventive Safety for Unprotected Road Users (PROTECTOR), took place from 2000 until 2003. The focus of this project was to develop visible light based pedestrian detection systems that could asses risk and, if necessary, alert the driver. The project used a system based approach to locate and identify pedestrians, track them and determine the risk of a collision. One notable aspect of this project was the use of over forty minutes of real-world traffic video to test the algorithms. While the researches admit that considerably more work needs to be done on the system, they did make many technological gains. [4]

PROTECTOR's successor, Sensors and System Architecture for Vulnerable Road Users Protection (SAVE-U) 2002-2005, built on some of the research from PROTECTOR but greatly enhanced the pedestrian detection and tracking system by adding a network of radar sensors and an infrared camera to the visible light sensor. The SAVE -U system uses multi-level sensor fusion in order to initially reduce the amount of image processing necessary by identifying regions of interest (ROIs) with the radar and, later the sensor fusion produces data to positively identify VRUs for tracking and characteristic determination. [5]

2

Finally, the DARPA sponsored Grand and Urban Challenges have spurred the production of a number of practical tools for autonomous vehicle navigation and control. The Urban Challenge, in particular, required the development of autonomous vehicles that could follow traffic laws, interact safely and correctly with other vehicles and navigate in an urban environment in the same manner that a human would. While this challenge did not directly involve any sort of pedestrian protection, the technology that was developed is a very good compliment to a pedestrian protection system. [6]

Pedestrian Detection and Classification

Researchers have developed an astonishing number of methods for pedestrian detection that use different sensors (or sets of sensors) with varying degrees of success. The most common methods use either visible or infrared light sensors – alone or in stereo pairs– or they pair light sensors with a range finder, such as RADAR, sonar or a laser scanner.

Visible Light

Visible light appears to be a logical method for pedestrian detection. Such cameras are common, relatively inexpensive and visible light is the means by which a human driver would detect and avoid pedestrians. However, extracting information from the images can be a complex, time consuming and inexact task and although many methods for detecting pedestrians have been developed, no single method stands out as superior in all conditions.

Locating pedestrians based on the unique shape or features of a human is widespread method of pedestrian detection. Papageorgiou and Poggio [7] use Haar wavelet features classified via a Support Vector Machine (SVM) to locate faces, people and cars in visible light images. Their system was integrated with a stereo obstacle detection system for use with the DaimlerChrysler Urban Traffic Assistant (UTA). Havasi *et al* [8] use the symmetry of human legs to locate humans in an image, while Abramson and Steux [9] use the diagonal orientation of human legs to find people in a single image, albeit with a higher rate of false positives.

Another common technique is to use the stereo disparity in the visible light images from two cameras to locate pedestrians. Gandhi and Trivedi [10] use a pair of omni directional cameras to locate obstacle in front of and to the side of a vehicle. Grubb *et al* [11] use stereo vision to locate obstacles and then obstacle size and SVMs to classify the obstacles as pedestrians. This technique can also be used for preprocessing to locate ROIs.

Finally, some researches have used the periodic motion of a pedestrians gait to identify the pedestrian. Niyogi and Adelson [12] look for a characteristic "braided" pattern from the movement of humans, from this they can identify individuals based on their distinctive gait. Cutler and Davis [13] use periodic motion to classify moving objects from both static and moving cameras.

Once a potential pedestrian is located in an image, most researches use additional algorithms to verify the nature of the potential pedestrians. Munder and Gavrila [14] looked at multiple feature-classifier combinations; comparing Haar Wavelets and local receptive fields (LRF) as well Support Vector Machines (SVMs), feed-forward networks and k-nearest neighbor classifiers to find combinations of classifiers that reliably detect pedestrians. They determined that a combination of SVMs with LRF worked best, though a boosted cascade of Haar Wavelets produced competitive results at a much lower computational cost.

Infrared

Both near and far infrared (NIR and FIR) sensors have been used for pedestrian detection and classification. Infrared sensors, especially FIR, reduce much of the complexity of visible light images as they separate warm objects from neutral temperature background clutter. Humans and the engine blocks of moving vehicles are usually the warmest objects in any environment likely to be encountered by vehicles. However, under extremely warm conditions, infrared sensors can lose some to all of their sensing capability as there is little difference between a human's body heat and the background temperature.

Infrared sensors are a significant improvement to visible light cameras at night. FIR requires no illumination and the generally cooler nighttime temperatures increase the contrast between a warm human and the background environment. FIR also does not get 'dazzled' or washed out by approaching head lights or other sudden change in lighting the way human eyes or visible light cameras do. While infrared cameras are usually more expensive then visible light cameras the cost has decreased considerably over the years, to the point that some luxury vehicles are starting to offer FIR imaging. [15]

One common method of pedestrian detection with infrared cameras is by human shape recognition. Broggi *et al* [16] use a single frame from an infrared camera to locate pedestrians by looking for symmetric, warm shapes with certain aspect ratios. Xu *et al* [17] use SVM to locate pedestrians in the images from a single night vision camera. They then track the pedestrian using mean shift tracking and Kalmen filter prediction. Meis *et al* [18] use simple pixel classification for head detection in addition to body classification in order to improve precision.

Another method that is becoming more common is to detect pedestrians in infrared videos by locating motion that is inconsistent with the background. Liu and Fujimura [19] use stereo infrared

cameras to detect obstacles and then examine motion that is independent from the background in order to classify obstacles as pedestrians; this method could be combined with the more traditional shape/texture based detection.

Sensor Fusion

Sensor fusion has been used frequently and with a great deal of success. The combination of sensors that collect data from different spectrums is often able to locate pedestrians more consistently, accurately and with significantly less computational cost then a single type of sensor. The combinations of sensors increase the equipment cost but greatly enhance that reliability of a system. Almost every autonomous vehicle that has been developed to the point of life-like testing has utilized multiple sensors and sensor fusion for navigation [6].

As previously mentioned, a frequent paring of sensors is a range sensor with a light sensor. The range sensor is used to detect obstacles, the locations of which are then mapped to the light based images as regions of interest (ROIs). By focusing on the ROIs, the amount of image processing needed for pedestrian detection is greatly reduced and the system can respond more quickly and with greater accuracy. Range sensors can also be used to calibrate pedestrian size when pedestrian detection is done via shape recognition. Milch and Behrens [20] use RADAR and vision to identify potential pedestrians and verify their nature using statistical shape models. Fang *et al* [21] use visible light and FIR to find and inspect ROIs and also suggest new features for segmentation which take advantage of the distinctive properties of FIR. Scheunert *et al* [22] use a laser scanner for ROI identification in an infrared image. They then combine the sensor information via parallel Kalman filters in order to sense and track multiple objects. LIDAR was used by Szarvas *et al* [23] to find ROIs in visible light images which were then analyzed with convolution neural networks to classify images of pedestrians.

Other researchers have used a combination of infrared and visible sensors. This combination creates a system that is reliable in a larger range of environments. Information can be collected from both sensors under normal conditions while in low light conditions the infrared camera provides the majority of the information and in extreme weather environments the visible light camera would be the main source of data.

Han and Bhanu [24] use color and infrared video to extract human silhouettes from a scene using image registration; silhouette extraction is often a precursor step to human recognition via gait identification. While the infrared video generally provides better human silhouettes, it was occasionally unreliable because the temperature of parts of the human body became close to that of the surrounding environment.

Pedestrian Motion Prediction

Predicting the motion of a pedestrian or another obstacle can be a challenging task. Methods for predicting pedestrian behavior are dependent upon the intended application scenario. For instance, a control system for a quickly moving vehicle on a straight road would only have the option to slow or stop the vehicle to avoid pedestrians and would thus only require a simple description about the projected path of the pedestrian – if the path was projected into the road and when. However, a control system for a robot navigating around a less structured environment, such as a parking lot, might need to plan a path around multiple moving people and would thus need to make more detailed and long term predictions about the future motion of the people. However, given that a large percentage of pedestrians involved in accidents are inebriated, one wonders about the accuracy of detailed models presumably based on sober subjects.

Linear Estimation

Linear estimators track an obstacle for a few sensor cycles to determine the obstacle's velocity. Future movement is predicted to continue at the same speed and direction as the previously recorded motion.

Tsuji *et al* [25] estimate the relative movement vector, between the pedestrian and the vehicle, based on principle component analysis. As the vehicle is, in this case, confined to a road lane where motion planning will consist only of speed changes along the road, elaborate motion prediction models for the vehicle to navigate around are not necessary. In addition, for vehicles traveling at high speeds the time between identification of a pedestrian and a possible collision are not usually long enough for the pedestrian to significantly change his or her speed or direction.

Modeling or Clustering

Modeling or clustering methods are two stage learning techniques. Motion patterns are learned during the first, observational, stage – this stage can be performed once and the results used repeatedly during operation. Later the motion of new obstacles is predicted based on which pattern it resembles.

Large *et al* [26] use a two stage motion predictor; the first stage monitors moving obstacles in a predefined environment and classifies the results into clusters of typical motion patterns. The second stage estimates the future motion of new obstacles based on the cluster with the maximum likelihood of the obstacle's initial movements.

Bennewitz *et al* [27] learn typical motion patterns of people using an EM-algorithm to cluster trajectories. Kalman filtering is used to track each person in the environment. They then weight an occupancy grid, with the probability of each cell being occupied, and use A* to plan a path.

Zhu [28] uses Hidden Markov Models to computational classify the motion of moving obstacles as constant velocity, random motion or intentional motion. Wakim *et al* [29] also use a Hidden Markov Model but with the four motion states: run, jog, walk and static. They use the current state to estimate future movement and changes in state.

Schulz *et al* [30] track multiple moving obstacles using motion models and sample-based Joint Probabilistic Data Association Filters, which do not require a Gaussian distribution, to associate detected features with tracked obstacles.

Shimizu and Poggio [31] use Haar wavelet features from sequences of images of walking people walking to train SVMs to estimate the walking direction of people. The actual estimation of walking direction can be performed with a single image.

Discrete Choice

Antonini *et al* [32] propose a discrete choice model to predict a pedestrian's future speed and direction based on data gathered from video of actual pedestrians. This model assumes that most pedestrians move directly toward their destination with predictable changes in speed or direction. Antonini and Bierlaire [33] later update their model to take pedestrian interaction into consideration.

Stationary Obstacle Avoidance/ Path Planning

There are two main types of configuration space path planers, topological and metric. Topological path planning uses landmarks to direct a vehicle. As the environment assumed for this project is not well enough know for the vehicle to navigate via landmarks, this review will focus on metric path planning, which can easily be broken down into sub goals for short term path planning.

There are numerous types of metric path planners, the most common of these, that are applicable for pedestrian avoidance with limited knowledge of the environment, are roadmap, cell decomposition and artificial potential field methods. Roadmap and cell decomposition path planners generally consist of two stages, representing the environment in configuration space and then an algorithm to determine the best path or roadmap through that space.

Roadmap Methods

There are several commonly used roadmap representation methods. Using the generalized Voronoi diagram, points are found that are equidistant from nearby obstacles. A path along these points, directed towards the goal location, is an extremely conservative means of obstacle avoidance. Ó'Dúnlaing and Yap [34] described the method of retracting free space into a Voronoi diagram where the vehicle is a disc. Latombe [35] gives a good general description of the Voronoi diagram algorithm and possible variations. Choset and Burdick [36] define the hierarchical generalized Voronoi diagram and show how it can be used for exploration of an unknown environment.

Unlike that Voronoi diagram which maximizes the path's distance from obstacles, a Visibility graph (or shortest-path roadmap) creates a path that hugs the vertices of obstacles in order to find the shortest path from the starting location to the goal. As the name alludes, a Visibility graph creates path segments from a vertex of an obstacle to all other obstacle vertices that are visible from the initial vertex. The path planning algorithm will then select the shortest set of segments to reach the goal. Nilsson first introduced this idea in 1969 [37]. Both Edelsbrunner [38] and Latombe [35] provide a through description. Oommen *et al* [39] use visibility graphs for robot navigation in an unexplored environment.

After the roadmap is created using one of either these methods, or the countless other methods mentioned in literature, a variety of algorithms exist to find the best route along the roadmap to the goal. The choice of these algorithms depends greatly on the amount and reliability of the information available to the path planner. Dijkstra's algorithm works by finding the lowest cost path from the initial stare to a signal state, assuming that all path costs are non negative. This algorithm does not require a heuristic function to predict the cost from future states to the goal state. Urdiales *et al* [40] use Dijkstra's algorithm to find the shortest path within a multi-level path planning algorithm. Qin *et al* [41] use a particle swarm optimization algorithm after Dijkstra's algorithm to find an optimal path.

A*, which is an extension of Dijkstra's algorithm, is often considered one of the best general planners. However to use A*, the path planner must have an admissible heuristic function to estimate (without overestimating) the cost to move from every state to the goal state. As the heuristic function is often not available for unexplored territory, this algorithm, while powerful, is limited in its application and can often only be used for path planning in the local region of an unknown environment. Alexopoulos and Griffin [42] use a variation of A* to find a collision free path among moving obstacles. Oriolo *et al* [43] use iterative applications of A* to generate local paths in an unknown environment.

Cell Decomposition

The environment can be also be broken down, via cell decomposition, into a grid of either regular or irregular grid elements. Depending on the technique used to create the grid, this method can be referred to as fixed, adaptive, approximate variable-cell or Quadtree cell decomposition, certainty grids or occupancy grids. For basic fixed cell decomposition, a Cartesian grid is superimposed on the environment. Grid elements are considered occupied if an obstacle resides in any part of the grid element; the path planning algorithm finds a path through unoccupied cells. Lozano-Perez [44] use cell decomposition for automatic planning of manipulator movement. Moravec and Elfes [45] use wide angle sonar to map an area and classify regions as empty, occupied or unknown. Thrun [46] uses data from multiple robots to create a three-dimensional map using occupancy grids. Jigong *et al* [47] use cell decomposition along with LGODAM to plan a path while avoiding obstacle traps.

For irregular grids, if an obstacle falls in part of a grid element, that element is divided up into smaller and smaller segments until each segment is either fully occupied by an obstacle or completely free. Zhu and Latombe [48] use constraint reformulation with hierarchical approximate cell decomposition to reduce the amount of area that contains a mix of occupied and unoccupied space. They also modify the planner to take advantage of information about previously made mistakes.

Artificial Potential Field Methods

Potential Field planners create a field or a gradient throughout the environment based on an attractive force exerted by the goal location and a localized repulsive force created by obstacles that should be avoided. The robot is then treated as a point under the influence of this field and is smoothly guided to the goal. O. Khatib [49] first introduce the artificial potential field concept and use it to control a manipulator in a complex environment. Borenstein and Koren [50] produce a virtual force field by combining certainty grids with an artificial potential field. Hwang and Ahuja [51] create a path by searching a graph created from the local minima of a potential field. Montano and Asensio [52] create an artificial potential field using a 3D laser range rotating sensor and show its usefulness on basic tasks such as avoiding obstacles or following walls. Batavia and Nourbakhsh [53] use a global potential field to perform the planning and navigation of a personal robot.

M. Khatib *et al* [54] introduce the rotation and task potential fields, which they refer to as the extended potential field. The rotation potential field takes the vehicles orientation into account when calculating an obstacle's repulsive field, in this was a vehicles traveling parallel to an obstacle would not suffer from the same repulsion as a vehicle directly approaching an obstacle. The task potential field allows the vehicle to ignore the repulsive fields of obstacle that it will not be approaching while completing its tasks.

Borenstein and Koren [55] introduce a variation on the artificial potential field method and the occupancy grid method; the Vector Field Histogram. The vector field histogram uses polar coordinates to prevent the vehicle from assuming trajectories that will approach obstacles while directing it to the goal. This method allows vehicles to be directed down narrow corridors or between close obstacles, when this is the shortest path, a path that is usually avoided with traditional potential field methods. Later, Ulrich and Borenstein introduce VHF+ [56], to improve reliability and smooth the robot trajectories, and VHF* [57], a combination of VHF+ and A*, to deal with traps that arise from typical short term planning methods.

Collision Prediction and Dynamic Obstacle Avoidance

The ability to avoid moving obstacles is a necessary for robots that must perform tasks in environments that contain vehicles, people or other non-stationary objects. Dynamic obstacle avoidance, while navigating to a goal, is a rapidly growing field due to the increasing number of situations where mobile robots or other autonomous systems are present. Numerous papers and a few books have been published on this subject; however no single method appears to be universally preferred, perhaps due in part to the wide variety of environments and applications for which autonomous or semi-autonomous robots are being used.

Adapted Static Obstacle Avoidance

Velocity Control

One of the simplest forms of motion planning in a dynamic environment involves generating a path among any static obstacles using traditional path planning algorithms (see previous section on static path planning) and then modifying the robot's velocity along that path, in real-time, to avoid dynamic obstacles. While this method is often successful at reaching the goal without encountering an obstacle, it cannot be guaranteed to find a time optimal path and in certain situations proves unable to avoid the dynamic obstacles. It does however reduce the computational time needed to determine a path for the robot as dynamic obstacles that do not intersect the pre-planned path can be ignored and obstacles that do interest the path need only be located a short time in advance and considered only in terms of speed adjustment.

Kant and Zucker [58 present this method and use modified velocity profiles along an original static path which was generated via a visibility graph approach. Lee and Lee [59], and Fujimura and Samet [60] also use a combination of velocity control along a visibility graph generated path.

Re-planning

Another method that uses static obstacles involves planning a path, using a very fast algorithm, around static obstacles and the present location of dynamic obstacle (treating them as static). As time progresses new paths are planned to take into account any change in the environment. While these paths are not always optimal, due to the continuous re-planning, they are well suited for environments where dynamic obstacles move infrequently or only small distances, such as an office where a chair may be shifted or a draw opened. Konolige [61] uses a gradient field to locally evaluate paths and determine if they are obstacle free. Oriolo *et al* [62] generate local paths within an explored area and terminate robot motion for unexpected (usually dynamic) obstacles. Fujimori *et al* [63] use adaptive navigation to detect and avoid obstacles in real time while respecting the dynamic limitations of the robot.

Obstacle Avoidance

Zhuang *et al* [64] use a path planner with a Visibility graph-like obstacle avoidance scheme to follow a very direct path to a goal. A robot, under this algorithm, follows a straight path to a goal (without any static obstacle pre-planning). When the algorithm detects an object within the current planning window the algorithm determines if it is a static or dynamic obstacle. If the obstacle is static, the algorithm plots sub-goal(s) to allow the robot to efficiently circumnavigate the obstacle and return to its original path. If the obstacle is dynamic then the algorithm uses auto-regression to predict the obstacles future position and the robot circumnavigates that position, updating its path in real time.

While this method does not always find a time or distance optimal path, no global knowledge is required which drastically reduces computation time. The short-term planning limit also makes the path planner more flexible for unknown and poorly know obstacle dynamics. However, modifications would have to be made to adapt the planner for uncertainty in the location of immediate obstacles.

Dynamic Obstacle Avoidance

Path Planning with a Time Dimension

Fraichard [65] presents a way to plan trajectories in a dynamic workspace which he entitles the 'state-time space' approach. If the current position of all static and dynamic obstacles and the velocity and acceleration of all dynamic obstacles is fully know before navigation, three dimensional path planners can plot an obstacle free path through 'state-time space,' in which the environment at each time step is treated as a two dimensional plane with time as a third dimension. Depending on the path planning algorithm used with this method, an optimal path to the goal (if one exists) can be guaranteed to be found.

However, the limitations on this method make it difficult to apply to real-world scenarios. While, as previously mentioned, methods exits to predict future (specifically) human movement none of these methods has nearly the certainty of performance that would be needed in order to use the 'state-time space' approach with its fullest advantage. In addition, the cost is prohibitive to completely up-date the path as changes in the velocity of obstacles are detected in the three-dimensional world.

Kindel *et al* [66] apply kinematic and dynamic constraints to the robot space-time planning and apply their results to an experimental robot with an overhead vision system.

Yu and Su [67] use a variation of 'state-time space' planning but limit the region of planning by focusing on "observation space," the area that the robot can sense, and "work space" the obstacles that are close to the robot. They also make extensive use of path repair algorithms to deal with dynamic obstacles and the inability to completely predict their future movement.

Genetic and Evolutionary Algorithms

Wang *et al* [68] use genetic algorithms to generate a path around static obstacles and the predicted collision points of dynamic obstacles based on a polygon representation of the obstacles. They reduce the calculation time needed for off-line planning and path re-calculation by considering only the vertices of obstacles using vertex++.

Xiao *et al* [69,70] develop and revise the Evolutionary Planner/ Navigator (EP/N). This planner/navigator can utilize specific environment knowledge to enhance its path planning performance.

Han *et al* [71] use genetic search algorithms to generate a goal directed dynamic path. Sugihara and Smith [72] use a genetic algorithm for path and trajectory planning. Their method is suitable for pre-planning as well as real-time motion planning.

Potential Field

The potential field method of obstacle avoidance, described by Borenstein and Koren [50] in the previous section on static obstacle avoidance, can be adapted for dynamic obstacle avoidance. O. Khatib [49]in his initial artificial potential field paper, surmises that a combination of high level (global) path planning with low level (local goal) planning could allow a manipulator to avoid moving obstacles.

Malik [73], on the other hand, develops the concept of the Extrapolated Potential Field, which predicts an obstacle's path and uses a time and distance weighting scheme to generate a path, for the robot to the goal, that avoids all obstacles. Similar to the static potential field planners, this method is usually quite fast at generating a path but will often miss potentially shorter routes between obstacles that are close together. As Borenstein and Koren mentioned for the static case, this type of planner is also subject

to oscillation and can run into problems with local minimum when confronted with a combination of static and dynamic obstacles.

The Linear Programming Navigation gradient method (LPN) was originally developed for static obstacles by Konolige [61], but Farinelli and Iocchi [74] modify this method for environments with dynamic obstacles. The LPN method uses numerical artificial potential fields that take both intrinsic (situational) and adjacency (movement) costs into account to compute a gradient using a generalization of the *wavefront* algorithm. The dynamic variation (LPN-DE) computes the projected motion of the obstacle and increases the weight of the region where the obstacle is predicted to travel to account for future movement.

Velocity Obstacles

Fiorini and Shiller[75] develop and Fiorini and Shiller [76]and Shiller *et al* [77] expand upon the notion of the velocity obstacle. Velocity obstacles are a first-order method of motion planning that use robot and obstacle velocities directly to avoid collisions in time-varying environments. This method computes a collision cone of robot velocities that will lead to probable collisions with an obstacle, based on the obstacle's current (and in later works) projected velocity. The velocity obstacle method takes the dynamic constraints of the robot into consideration to narrow down the field of potential robot velocities.

In later papers, Large *et al* [78] adapt the velocity obstacle concept to account for risk and long obstacles (such as hallway walls).

Yamamoto *et al* [79] apply the velocity obstacle concept to situations more likely to be encountered in the real world including obstacles that change velocity during sensor cycles and they also introduce the idea of a collision distance index to prioritize the avoidance of obstacles that are closer (and thus pose a more imminent threat) to the robot.

Dynamic Window/ Curvature Velocity Method

The dynamic window approach, which can also been used as a simplifying adaptation on other algorithms, reduces the complexity of path planning by only considering velocities that the robot can reach safely with in a short time interval. Using this method, all of the safe and reachable velocities of the robot make up the dynamic window, which is represented in velocity space. On its own, the Dynamic Window Method is best suited for static environments or environments that have few, slowly moving dynamic obstacles. However, it can be a very powerful tool when combined with other algorithms.

Fox *et al* [80] use the dynamic window approach to account for the robot's dynamic constraints and applied the algorithm to their robot RHINO. Brock and O. Khatib [81] propose the global dynamic

window approach to combine path planning with real-time obstacle avoidance in order safely navigate in a dynamic environment while approaching a goal.

Bibliography

1. "Traffic Safety Facts: 2006 Data, Pedestrians." *NHTSA's National Center for Statistics and Analysis.* 2007. 12 March 2008 <http://www.nhtsa.dot.gov/portal/site/nhtsa/menuitem.dfedd570f698cabbbf30811060008a0c>

2. Army News Services. "Army Sees 20-Percent Reduction in Accident Fatalities." *Veteran News and Information Services.* 16 Jan. 2006. 12 March 2008 <http://www.vnis.com/story.cfm?textnewsid=2244>

3. [Online]. Available: http://www.worldbank.org/html/fpd/transport/roads/safety.htm

4. Gavrila, D.M.; Giebel, J., Munder, S., "Vision-based pedestrian detection: the PROTECTOR system," *Intelligent Vehicles Symposium, 2004 IEEE* , vol., no., pp. 13-18 June 2004

5. Töns, M., Doerfler, R., Meinecke, M., Obojski, M.A., "Radar Sensors and Sensor Platform used for Pedestrian Protection in the EC-funded project SAVE-U," *Intelligent Vehicles Symposium, 2004 IEEE* vol., no., pp. 813-818, June 2004

6. [Online]. Available: http://www.darpa.mil/GRANDCHALLENGE

7. Papageorgiou, C. and Poggio, T., "A trainable system for object detection," *International Journal on Computer Vision*, vol. 38, pp. 15, 2000.

8. Havasi, L., Szlávik, Z., and Szirányi T.,"Pedestrian detection using derived third-order symmetry of legs," *Proceedings of the International Conference Computer Vision Graph.*, 2004.

9. Abramson, Y.; Steux, B., "Hardware-friendly pedestrian detection and impact prediction," *Intelligent Vehicles Symposium, 2004 IEEE* , vol., no., pp. 590-595, 14-17 June 2004

10. Gandhi, T.; Trivedi, M.M., "Vehicle mounted wide FOV stereo for traffic and pedestrian detection," *Image Processing, 2005. ICIP 2005. IEEE International Conference on* , vol.2, no., pp. II-121-4, 11-14 Sept. 2005

11. Grubb, G.; Zelinsky, A.; Nilsson, L.; Rilbe, M., "3D vision sensing for improved pedestrian safety," *Intelligent Vehicles Symposium, 2004 IEEE* , vol., no., pp. 19-24, 14-17 June 2004

12. Niyogi, S.A.; Adelson, E.H., "Analyzing and recognizing walking figures in XYT," *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on* , vol., no., pp.469-474, 21-23 Jun 1994

13. Cutler, R.; Davis, L.S., "Robust real-time periodic motion detection, analysis, and applications," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* , vol.22, no.8, pp.781-796, Aug 2000

14. Munder, S.; Gavrila, D.M., "An Experimental Study on Pedestrian Classification," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* , vol.28, no.11, pp.1863-1868, Nov. 2006

15. [Online]. Available: http://www.bmwusa.com/Standard/Content/Vehicles/2008/5/535xiSedan/Features_and_Specs /535xiSedanPkgsandOptions.aspx

16. Broggi, A.; Fascioli, A.; Carletti, M.; Graf, T.; Meinecke, M., "A multi-resolution approach for infrared vision-based pedestrian detection," *Intelligent Vehicles Symposium, 2004 IEEE* , vol., no., pp. 7-12, 14-17 June 2004

17. Xu, F.; Liu X.; Fujimura, K., "Pedestrian detection and tracking with night vision," *Intelligent Transportation Systems, IEEE Transactions on* , vol.6, no.1, pp. 63-71, March 2005

18. Meis, U.; Oberlander, M.; Ritter, W., "Reinforcing the reliability of pedestrian detection in far-infrared sensing," *Intelligent Vehicles Symposium, 2004 IEEE* , vol., no., pp. 779-783, 14-17 June 2004

19. Liu, X.; Fujimura, K., "Pedestrian detection using stereo night vision," *Vehicular Technology, IEEE Transactions on* , vol.53, no.6, pp. 1657-1665, Nov. 2004

20. Milch, S.; Behrens, M., *Pedestrian Detection With Radar and Computer Vision* [online] Available: http://www.smart-microwave-sensors.de/Pedestrian_Detection.pdf.

21. Fang, Y.; Yamada, K.; Ninomiya, Y.; Horn, B.; Masaki, I., "Comparison between infrared-image-based and visible-image-based approaches for pedestrian detection," *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE* , vol., no., pp. 505-510, 9-11 June 2003

22. Scheunert, U.; Cramer, H.; Fardi, B.; Wanielik, G., "Multi sensor based tracking of pedestrians: a survey of suitable movement models," *Intelligent Vehicles Symposium, 2004 IEEE* , vol., no., pp. 774-778, 14-17 June 2004

23. Szarvas, M.; Yoshizawa, A.; Yamamoto, M.; Ogata, J., "Pedestrian detection with convolutional neural networks," *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE* , vol., no., pp. 224-229, 6-8 June 2005

24. Han, J., Bhanu, B., "Detecting Moving Humans Using Color and Infrared Video," *IEEE Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2003.

25. Tsuji, T.; Hattori, H.; Watanabe, M.; Nagaoka, N., "Development of night-vision system," *Intelligent Transportation Systems, IEEE Transactions on* , vol.3, no.3, pp. 203-209, Sep 2002

26. Large, F.; Vasquez, D.; Fraichard, T.; Laugier, C., "Avoiding cars and pedestrians using velocity obstacles and motion prediction," *Intelligent Vehicles Symposium, 2004 IEEE* , vol., no., pp. 375-

379, 14-17 June 2004

27. Bennewitz, M.; Burgard, W.; Thrun, S., "Adapting navigation strategies using motions patterns of people," *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on* , vol.2, no., pp. 2000-2005 vol.2, 14-19 Sept. 2003

28. Zhu, Q., "Hidden Markov model for dynamic obstacle avoidance of mobile robot navigation," *Robotics and Automation, IEEE Transactions on* , vol.7, no.3, pp.390-397, Jun 1991

29. Wakim, C.F.; Capperon, S.; Oksman, J., "A Markovian model of pedestrian behavior," *Systems, Man and Cybernetics, 2004 IEEE International Conference on* , vol.4, no., pp. 4028-4033 vol.4, 10-13 Oct. 2004

30. Schulz, D.; Burgard, W.; Fox, D.; Cremers, A.B., "Tracking multiple moving objects with a mobile robot," *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on* , vol.1, no., pp. I-371-I-377 vol.1, 2001

31. Shimizu, H.; Poggio, T., "Direction estimation of pedestrian from multiple still images," *Intelligent Vehicles Symposium, 2004 IEEE* , vol., no., pp. 596-600, 14-17 June 2004

32. Antonini, G.; Venegas, S.; Thiran, J.P.; Bierlaire, M., "A discrete choice pedestrian behavior model for pedestrian detection in visual tracking systems," *Proceedings on Advanced Concepts in Intelligent Vision Systems.* Brussels, Belgium, Sep. 2004

33. Antonini, G.; Bierlaire, M "Capturing interactions in pedestrian walking behavior in a discrete choice framework," *Transportation Research, Part B*, vol. 40, pp. 667, 2006.

34. O'D´unlaing, C. and Yap, C. K. (1982). A retraction method for planning the motion of a disc. *Journal of Algorithms*, 6, 104–111.

35. Latombe, J. C., *Robot Motion Planning* Boston: Kluwer, 1991.

36. Choset, H.; Burdick, J., "Sensor-based exploration: The hierarchical generalized Voronoi graph," *International Journal of Robotics Research*, vol. 19, no. 2, pp. 96-125, 2000

37. Nilsson, N.J, "A mobile automaton: an application of artificial intelligence techniques," *Proceedings IJCAI-I,* Washington, DC, 1969.

38. Edelsbrunner, H., Algorithms in Combinatorial Geometry. Springer-Verlag, Berlin, Heidelberg, New York, London, Paris, Tokyo, 1988. Berlin, Heidelberg, New York, London, Paris, Tokyo, 1988.

39. Oommen, B.; Iyengar, S.; Rao, N.; Kashyap, R., "Robot navigation in unknown terrains using learned visibility graphs. Part I: The disjoint convex obstacle case," *Robotics and Automation, IEEE Journal of [legacy, pre - 1988]* , vol.3, no.6, pp. 672-681, Dec 1987

40. Urdiales, C.; Bandera, A.; Arrebola, F.; Sandoval, F., "Multi-level path planning algorithm for autonomous robots," *Electronics Letters* , vol.34, no.2, pp.223-224, 22 Jan 1998

41. Qin, Y.; Sun, D.; Li, N.; CenY., "Path planning for mobile robot using the particle swarm optimization with mutation operator," *Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on* , vol.4, no., pp. 2473-2478 vol.4, 26-29 Aug. 2004

42. Alexopoulos, C.; Griffin, P.M., "Path planning for a mobile robot," *Systems, Man and Cybernetics, IEEE Transactions on* , vol.22, no.2, pp.318-322, Mar/Apr 1992

43. Oriolo, G.; Ulivi, G.; Vendittelli, M., "Real-time map building and navigation for autonomous robots in unknown environments," *Systems, Man, and Cybernetics, Part B, IEEE Transactions on,* vol.28, no.3, pp.316-333, Jun 1998

44. Lozano-Perez, T., "Automatic planning of manipulator transfer movements," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 11, pp. 681-698, 1981.

45. Moravec, H.; Elfes, A., "High resolution maps from wide angle sonar," *Robotics and Automation. Proceedings. 1985 IEEE International Conference on* , vol.2, no., pp. 116-121, Mar 1985

46. Thrun, S., "A probabilistic online mapping algorithm for teams of mobile robots," *International Journal of Robotics Research,* 20(5): 335-363, 2001.

47. Li Jigong,; Yiwei, Feng; Chaoqun, Zhu, "A Novel Path Planning Method Based on Certainty Grids Map For Mobile Robot," *Control Conference, 2007. CCC 2007. Chinese* , vol., no., pp.185-188, July 26 2007-June 31 2007

48. Zhu, D.J.; Latombe, J.-C., "New heuristic algorithms for efficient hierarchical path planning ," *Robotics and Automation, IEEE Transactions on* , vol.7, no.1, pp.9-20, Feb 1991

49. Khatib, O., "Real-time obstacle avoidance for manipulators and mobile robots," *Robotics and Automation. Proceedings. 1985 IEEE International Conference on* , vol.2, no., pp. 500-505, Mar 1985

50. Borenstein, J.; Koren, Y., "Real-time obstacle avoidance for fact mobile robots," *Systems, Man and Cybernetics, IEEE Transactions on* , vol.19, no.5, pp.1179-1187, Sep/Oct 1989

51. Hwang, Y.K.; Ahuja, N., "A potential field approach to path planning," *Robotics and Automation, IEEE Transactions on* , vol.8, no.1, pp.23-32, Feb 1992

52. Montano, L.; Asensio, J.R., "Real-time robot navigation in unstructured environments using a 3D laser rangefinder," *Intelligent Robots and Systems, 1997. IROS '97., Proceedings of the 1997 IEEE/RSJ International Conference on* , vol.2, no., pp.526-532 vol.2, 7-11 Sep 1997

53. Batavia, P.; Nourbakhsh, I., "Path Planning for the Cye Robot," *Proceedings of IROS 2000*, Vol. 1, October, 2000, pp. 15 - 20.

54. Khatib, M., Jaouni, H., Chatila, R., Laumod, J.P., "Dynamic Path Modification for Car-Like Nonholonomic Mobile Robots," *IEEE Int. Conf. on Robotics and Automation*, Albuquerque, USA, 1997.

55. Borenstein, J.; Koren, Y., "The vector field histogram-fast obstacle avoidance for mobile robots," *Robotics and Automation, IEEE Transactions on* , vol.7, no.3, pp.278-288, Jun 1991

56. Ulrich, I.; Borenstein, J., "VFH+: reliable obstacle avoidance for fast mobile robots," *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on* , vol.2, no., pp.1572-1577 vol.2, 16-20 May 1998

57. Ulrich, I.; Borenstein, J., "VFH[*]: local obstacle avoidance with look-ahead verification," *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on* , vol.3, no., pp.2505-2511 vol.3, 2000

58. Kant, K. and Zucker, S. W. (1986). "Toward efficient trajectory planning: The path-velocity decomposition," *International Journal of Robotics Research*, 5(3): 72-89.

59. Lee, B., and Lee, C. 1987. "Collision-free motion planning of two robots," *Systems, Man and Cybernetics, IEEE Transactions on* 17(1):21-32.

60. Fujimura, K., and Samet, H. 1993. "Planning a time-minimal motion among moving obstacles," *Algorithmica* 10:41-63.

61. Konolige, K., "A gradient method for real time robot control," *Intelligent Robots and Systems, 2000. (IROS 2000). Proceedings. 2000 IEEE/RSJ International Conference on* , vol.1, no., pp.639-646 vol.1, 2000

62. Oriolo, G.; Ulivi, G.; Vendittelli, M., "Real-time map building and navigation for autonomous robots in unknown environments," *Systems, Man, and Cybernetics, Part B, IEEE Transactions on,* vol.28, no.3, pp.316-333, Jun 1998

63. Fujimori, A.; Nikiforuk, P.N.; Gupta, M.M., "Adaptive navigation of mobile robots with obstacle avoidance," *Robotics and Automation, IEEE Transactions on* , vol.13, no.4, pp.596-601, Aug 1997

64. Zhuang, H.; Li, H.; Du S., "Real-time Path Planning of Mobile Robots in Dynamic Uncertain Environment," *Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on*, vol.2, no., pp. 8886-8891, 21-23 June 2006

65. Fraichard, T., "Trajectory Planning Amidst Moving Obstacles: Path-Velocity Decomposition Revisited," *Journal of the Brazilian Computer Society* [online]. 1998, vol. 4, no. 3 [cited 2008-03-13].

66. Kindel, R.; Hsu, D.; Latombe, J.-C.; Rock, S., "Kinodynamic motion planning amidst moving obstacles," *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on* , vol.1, no., pp.537-543 vol.1, 2000

67. Yu, H.; Su, T., "A destination driven navigator with dynamic obstacle motion prediction," *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on* , vol.3, no., pp. 2692-2697 vol.3, 2001

68. Wang Y.; Sillitoe, I.P.W.; Mulvaney, D.J., "Mobile Robot Path Planning in Dynamic Environments," *Robotics and Automation, 2007 IEEE International Conference on* , vol., no., pp.71-76, 10-14 April 2007

69. Xiao, J., "Evolutionary planner/navigator in a mobile robot environment," *Handbook of Evolutionary Computation*, , New York: Oxford Univ. Press and Institute of Physics, 1997

70. Xiao, J.; Michalewicz, Z.; Zhang, L., "Evolutionary planner/navigator: Operator performance and self-tuning," *Proceedings 3rd IEEE International Conference Evolutionary Computation* Nagoya, Japan, pp. 366-371, May 1996.

71. Han, W.; Baek, S.; Kuc, T,, "Genetic algorithm based path planning and dynamic obstacle avoidance of mobile robots," *Systems, Man, and Cybernetics, 1997. 'Computational Cybernetics and Simulation'., 1997 IEEE International Conference on* , vol.3, no., pp.2747-2751 vol.3, 12-15 Oct 1997

72. Sugihara, K.; Smith, J., ``Genetic Algorithms for Adaptive Planning of Path and Trajectory of a Mobile Robot in 2D terrains", *IEICE Transactions Information and Systems*, vol. E82-D, no. 1, pp. 309--316, Jan. 1999.

73. Malik, W., "Motion planning of mobile robot in dynamic environment using potential field and roadmap based planner," Electronic Thesis [online] Available: http://txspace.tamu.edu/handle/1969.1/338?mode=full Texas A&M University, August 2003

74. Farinelli, A.; Iocchi, L., "Planning trajectories in dynamic environments using a gradient method." In *Proceedings of RoboCup Symposium* (Padua, Italy, 2003).

75. Fiorini, P.; Shiller, Z., "Motion planning in dynamic environments using the relative velocity paradigm," *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on* , vol., no., pp.560-565 vol.1, 2-6 May 1993

76. Fiorini, P.;Shiller, Z., "Motion planning in dynamic environments using velocity obstacles. *International Journal of Robotics Research,* 17(7):760-772, July 1998

77. Shiller, Z.; Large, F.; Sekhavat, S., "Motion planning in dynamic environments: obstacles moving along arbitrary trajectories," *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on* , vol.4, no., pp. 3716-3721 vol.4, 2001

78. Large, F.; Sekhavat, S.; Shiller, Z.; Laugier, C., "Towards real-time global motion planning in a dynamic environment using the NLVO concept," *Intelligent Robots and System, 2002. IEEE/RSJ International Conference on* , vol.1, no., pp. 607-612 vol.1, 2002

79. Yamamoto, M.; Shimada, M.; Mohri, A., "Online navigation of mobile robot under the existence of dynamically moving multiple obstacles," *Assembly and Task Planning, 2001, Proceedings of the IEEE International Symposium on* , vol., no., pp.13-18, 2001

80. Fox, D.; Burgard, W.; Thrun, S., "The dynamic window approach to collision avoidance," *Robotics & Automation Magazine, IEEE* , vol.4, no.1, pp.23-33, Mar 1997

81. Brock, O.; Khatib, O., "High-speed navigation using the global dynamic window approach," *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on* , vol.1, no., pp.341-346 vol.1, 1999