A Common Operator Control Unit Color Scheme for Mobile Robots

M. Shneier, R. Bostelman, J. S. Albus, W. Shackleford, T. Chang, T. Hong
Intelligent Systems Division
National Institute of Standards and Technology
Gaithersburg, MD

Abstract

The Intelligent Systems Division at the National Institute of Standards and Technology (NIST) has participated in the Defense Advanced Research Project Agency (DARPA) Learning Applied to Ground Robots (LAGR) project for the past 2 ½ years. In Phase 2 of the LAGR program, NIST was asked to provide a common operator control unit (OCU) color scheme for all LAGR teams to use. The color scheme simplifies the task of LAGR's evaluation team by providing a straightforward way to compare the performance of each of the teams using the different OCUs. During Phase 1, LAGR performers applied their own standards to the OCU color scheme and DARPA and other performers had a very difficult experience evaluating what the robot was computing based on stereo image, instrumented bumper, and inertial data.

NIST developed the color scheme based on real-world conventions and on the desire to accommodate as much of the teams' existing color schemes as possible. For example, typically red lights mean stop and green lights mean go for automobiles. This scheme was adopted by coloring obstacles red and traversable ground green in the new common color scheme. Red, green, blue (RGB) colors were produced for a variety of necessary parameters including: unknown regions, lethals, bumper hits, road/path, planned and traversed paths, goal, and waypoints. Also, vehicle modes were expressed such as: Normal Control, Aggressive, Backing, Stopped, and Manual modes.

The paper discusses the color scheme for ground robots developed for the LAGR Program.

I Introduction

The Operator Control Unit (OCU) for a mobile robot needs to display a lot of complex information about the state and planned actions of the vehicle. This includes displays from the robot's sensors, maps of what it knows about the world around it, traces of the path it has already traveled and predictions of the path it is planning to take, and information about obstacles, clear ground, and unseen regions. The information needs to be easy to understand even by people who have no understanding of the way the control system of the robot works, and should enable them to halt the vehicle only if it is about to take an action that will cause damage.

In order to display all the information in an understandable way, it is necessary to use color to represent the different types of region. There are no existing conventions on what colors to use and each OCU developer must decide how best to assign the colors. This is fine if the color scheme chosen is indeed easy to understand and if the people using the OCU are only dealing with a single robot. However, there is a growing need for the ability to control more than one robot, especially in the areas of urban search and rescue and bomb disposal robots, where a damaged or destroyed robot needs to be replaced as soon as possible.

There has been at least one effort to create a multi-robot OCU[1], called MOCU. The goal is to create an OCU that can control multiple vehicles of different types and from different manufacturers from a single OCU. The OCU is based on a core set of capabilities enhanced by modules that provide specific capabilities and communication protocols needed by each vehicle. The capabilities are defined at run time through configuration files. An impressive set of vehicles can be monitored simultaneously using MOCU. Unfortunately for the purposes of this paper, no standard color scheme is described for MOCU, so it was not possible to benefit from this approach.

Another paper does describe the color scheme used in its multi-vehicle OCU [2]. They describe the following assignments of colors, some of which are similar to the choices discussed below. The display a two-dimensional scrollable map of the search area marked with icons to indicate the relative positions of the robots, obstacles, target munitions and terrain features. Icons for robots change color to indicate the current status of the robot: green indicates operation within normal limits and red indicates a fault or a lack of progress. In the work described in this paper, multiple robot modes are described with different colors. Robot icons also indicate the heading of the robot and its position relative to domain objects and other robots. An obstacle detected by the robot or manually entered by the operator is displayed as a gray box. Areas that are indicated clear by the robot's IR sensors can optionally be colored blue on the operator's display, to distinguish those areas from unswept areas. If the robot detects steep parts of the terrain, the OCU can mark those areas yellow or orange, depending on the sensed inclination. As the robot moves, the clear areas under the munitions detector are shown in green, similar to what is used in our OCU. When a robot detects unexploded ordinance, its location is marked with a red circle on the OCU display. In the OCU described here, critical obstacles are also displayed in red.

The OCU developed in this work was oriented towards a somewhat different purpose. As part of the DARPA LAGR program [3], a large number of teams were tasked with developing learning algorithms to try to improve their performance in a series of field trial, held once a month. After the first phase of the program, NIST was asked to develop a common color scheme for the OCUs used by each team so that the evaluation team could reduce the need to learn the meaning of colors used by each team.

II The DARPA LAGR Program

The DARPA LAGR program [4] aims to develop algorithms that will enable a robotic vehicle to travel through complex terrain without having to rely on hand-tuned algorithms that only apply in limited environments. The goal is to enable the control system of the vehicle to learn which areas are traversable and how to avoid areas that are impassable or that limit the mobility of the vehicle. To accomplish this goal, the program provided small robotic vehicles to each of the participants (Figure 1). The vehicles are used by the teams to develop software. A separate LAGR Government Team, with an identical vehicle, conducts tests of the software each month.

The vehicle provided by DARPA is a small but very capable robot with substantial on-board processing capacity and a rich set of sensors. The sensors include two pairs of color cameras mounted on a turret on the front of the vehicle, a pair of infra-red range sensors (non-contact bumpers) on the front of the vehicle, and a physical bumper centered on the front wheels of the vehicle. For position sensing, the vehicle has a Global Positioning System (GPS) receiver, wheel encoders, and an inertial navigation system (INS). In addition, there are sensors for motor current, battery level, and temperature. There are four single-board computers on the vehicle, one for low-level vehicle control, one for each of the stereo camera pairs, and one for overall control of the vehicle. All processors use the Linux operating system. The vehicle has an internal Ethernet network connecting the processors, and a wireless Ethernet link to external processors.

The availability of range information from stereo vision enables the robot to navigate largely using the geometry of the scene. Sensor processing is aimed at determining where the vehicle is and what parts of the world around it are traversable. The robot can then plan a path over the traversable region to get to its goal.

When the vehicles were delivered, they came with a baseline control system and a baseline OCU. While some of the teams stuck with the baseline OCU, many developed their own OCU to provide better information for debugging and monitoring the robot. Since there were no standards for how the OCU should look, each of the teams developed dissimilar appearance models. As a result, when the Government evaluation team ran the monthly tests, they had to learn the conventions used by each OCU in order to monitor performance of the associated vehicle controller. This led them to request that NIST develop a common color scheme for Phase II of the program.

Figure 1. The robot used in the DARPA LAGR program.

III The Color Scheme

There is a large amount of literature on color and how to select color schemes for human-computer interfaces. Reference [5] provides a brief overview and links to more comprehensive resources. Many of these approaches make use of the color wheel and recommend selecting complementary or analogous colors. There are many tools to help in this selection [6]. This approach breaks down, however, when more than a small number of colors have to be selected and when certain colors have an accepted meaning in the application.

The way the color scheme for the LAGR OCUs was developed was to start out with a straw man proposal, which was sent out to the teams and to the DARPA Program Manager. Comments were received from a number of teams, which resulted in a revised scheme. This process was iterated until a scheme acceptable to all was developed. This may not be the best way to assign colors, but the intention was not to find the most pleasing scheme, but one that would be easy to interpret. Note that the color scheme was not intended to change the individual teams' OCU layout or content, only to ensure that the appearance of similar information in different OCUs would be consistent in meaning.

For the OCU, a large amount of information had to be represented simultaneously. Most of the teams displayed similar information, but allowance had to be made for extra features if they were required by even one of the teams. The information to be displayed included maps, which often included a low resolution long-range map and a higher resolution close-up map. These displays contain most of the information the color scheme needs to represent. Also displayed is vehicle mode (normal, aggressive, backing up, etc.). A third set of displays includes one or more images of what the vehicle is currently seeing. While there are often overlays on the images, so far the colors for these have not been standardized.

Maps need to display a wide range of information about the terrain and the planned and traversed paths. The colors assigned to the various features are shown in Figure 2 and 3. The same color scheme is used for both the low resolution and high resolution maps. Most colors are fixed, and refer to a single type of feature, but traversability ranges across a green-to-black spectrum based on how expensive it is to cover the associated terrain. Green means the vehicle can easily drive, while black means it is very difficult to drive. Note that in the Hue, Saturation, Luminance (HSL) color space, this requires changing only the Luminance component of the color, making for an easy mapping from cost to color. These costs, in conjunction with obstacles and bumper hits, are used by the planners to determine the optimal path to the goal. Comparing our scheme for representing steep slopes with that of [2], they have two fixed colors, orange and yellow for different degrees of steepness, while our scheme would assign smoothly-varying colors starting as green at the base of the slope and becoming darker as the slope gets steeper. This, we believe, gives more information about the true nature of the traversability of the slope.

	Traversability cost ranges from low = green (0,255,0) to high = black (0, 0, 0)
	Unknown regions = blue (0, 0, 255
	Lethal obstacles = red (255, 0, 0)
	Bumper hits = dark red (170, 0, 0)
	Road/path = light gray (215, 215, 215)
	High-level planned path = yellow (255,255,0)
	Low-level planned path = orange (255, 150, 0)
	Traversed path = dashed purple (255, 0, 255)
	Goal direction = white (255, 255, 255)
	Goal = white square (255, 255, 255)
	Waypoints = yellow squares on high level path (255,255,0)
	Camera FOV border = white (255, 255, 255)

Figure 2. The map colors and their interpretations.

The vehicle itself is displayed on the map, together with its field of view (FOV), planned path, and the path traversed so far. The planned path includes waypoints and an indication of the location of the goal. The straight-line path to the goal is also indicated. If the vehicle controller determines that it is traversing a road or path, this is displayed in gray.

The display of the vehicle changes color to indicate the current mode (Figure 4). Modes represent the status of the vehicle. Not all teams make use of all the modes, but simple ones, such as normal autonomous driving, stopped, and backing up are universal.

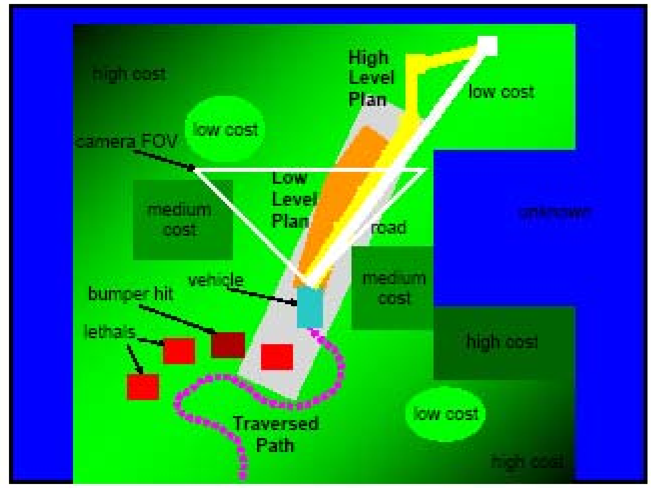

Figure 3. A schematic showing the meaning of the colors on the map.

	Normal Control = aqua (40, 200, 200)
	Aggressive = dark purple (130, 0, 130)
	Backing up = purple (255, 0, 255)
	Stopped = brown (150, 100, 50)
	Manual = light blue (120, 120, 255)

Figure 4. Vehicle modes displayed by coloring the vehicle on the map.

IV Implementation by LAGR Teams

Each of the LAGR teams was required to implement the scheme by the February 2007 test date. The common OCU Color Scheme code was sent to all teams in December, 2006. Implementation was only partially achieved, in that some teams did not have their OCUs ready in time, and some implemented only some aspects of the color scheme. NIST implemented the scheme (Figure 5 and Figure 6), showing both a high-resolution near-range map and a low-resolution, longer range map. In these figures, the vehicle believes that it is traversing a path, so the planned trajectory remains on the path until it ends and the vehicle has to enter unknown terrain to reach the goal.

SRI International originally developed their own color scheme, shown in Figure 7, which included only some of the new scheme features shown in Figure 2. There is no representation of the vehicle, and no waypoints, and the traversable terrain is shown in a single color. While the items represented have not changed, SRI implemented the full color scheme (Figure 8, by permission of SRI) so that the terrain is now shown with variation in traversal cost, obstacles are in red, and unknown regions in blue.

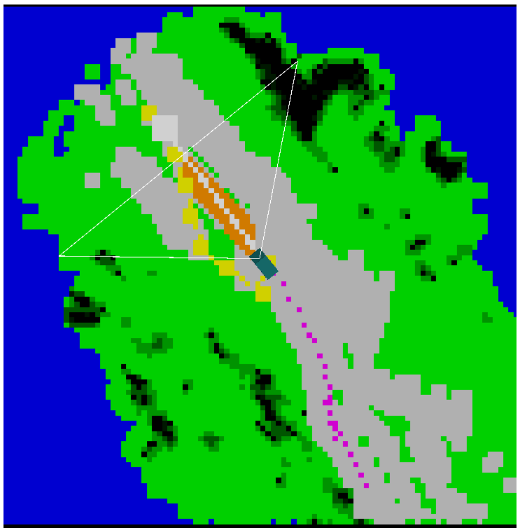

Figure 5. The NIST high resolution, short-range map.

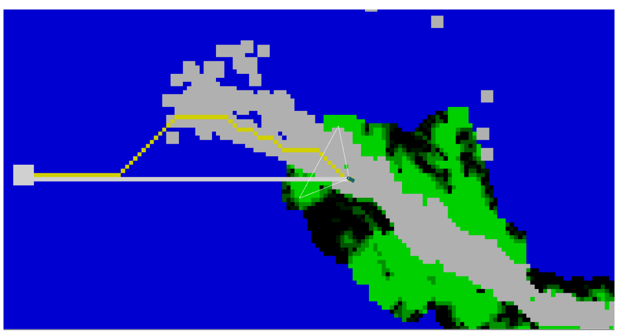

Figure 6. The NIST low resolution, long-range map.

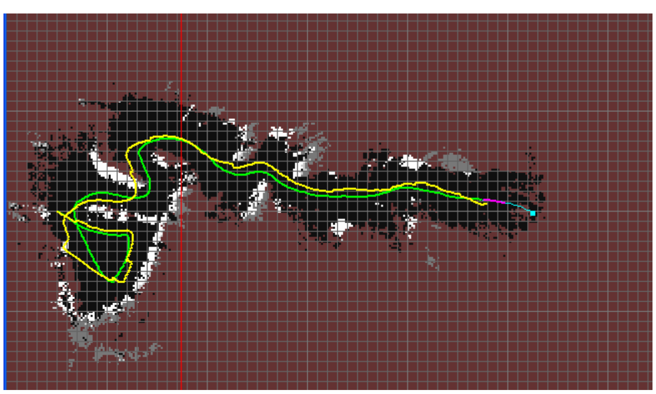

Figure 7. The old SRI OCU color scheme.

Another of the teams, Netscale Technologies, Inc., converted from their old, rather attractive, color scheme shown in Figure 9 for the high-level (low resolution) map and Figure 10 for the high resolution map. Their implementation of the common color scheme is shown in Figure 11 (low resolution) and Figure 12 (high resolution). Note that Netscale made other changes to what is represented in the OCU, so the maps are not directly comparable.

An interesting variant of the color scheme was implemented by the University of Pennsylvania. While it uses similar colors to the common color scheme, it uses pastel versions. For their scheme, the color correspondence to the map cost from low to high is shown in the color bar on the right of Figure 13 (by permission of U. Penn.). Green means safe to drive over, red marks obstacles, blue areas are unknown, and white indicates the planned path. This is in contrast to their old color scheme, shown in Figure 14. That color scheme was based on shades of gray. The higher the map cost, the darker the grayscale, and the lower the map cost, the whiter the color. In this scheme, white meant safe to drive over, dark gray (black) meant an obstacle, midgray was unknown, and blue indicated the planned path. Clearly, the new color scheme is closer to the standard, but it is not fully compliant.

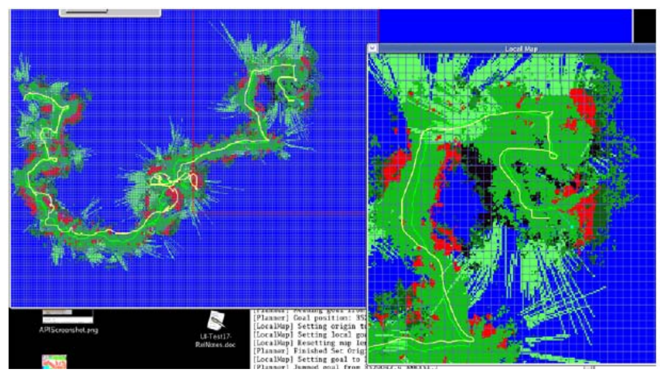

Figure 8. The new SRI OCU using the common color scheme.

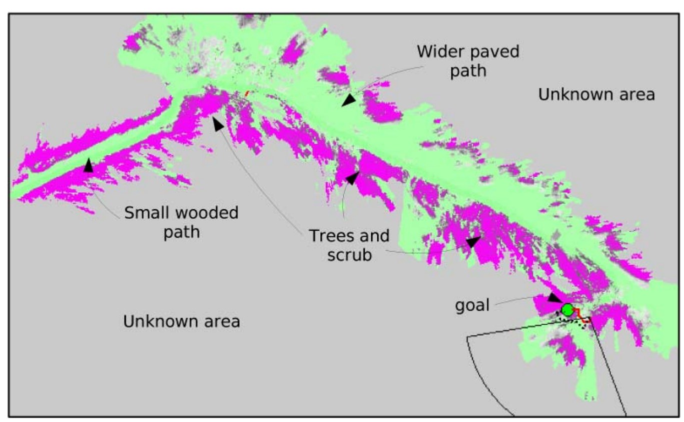

Figure 9. The old Netscale low resolution map color scheme (with permission from Netscale Technologies).

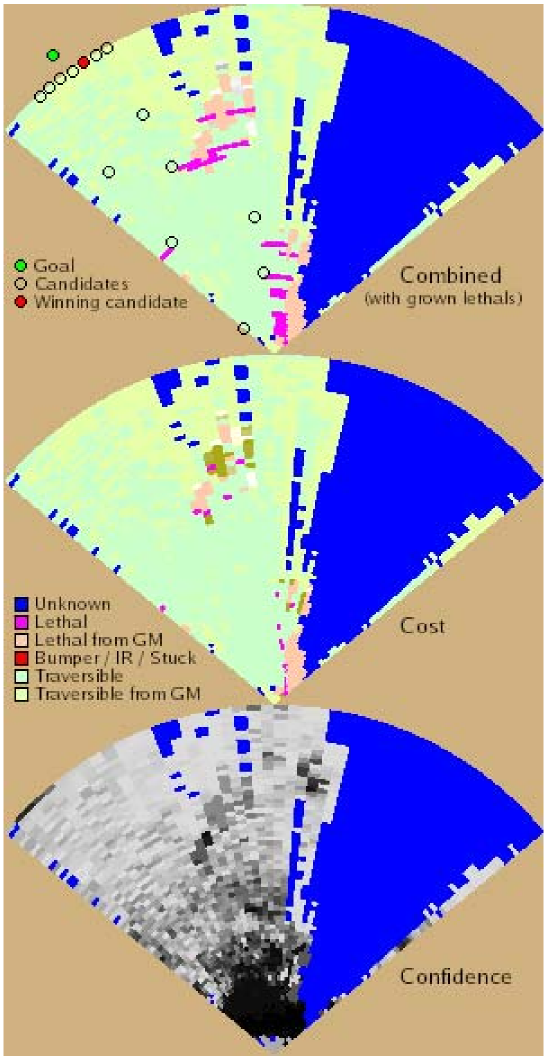

Figure 10. Old Netscale high-resolution maps.

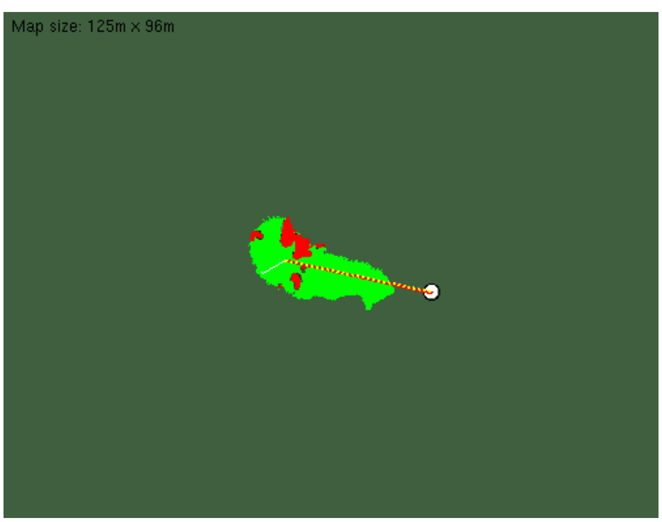

Figure 11. New Color Scheme for Netscale's low resolution maps.

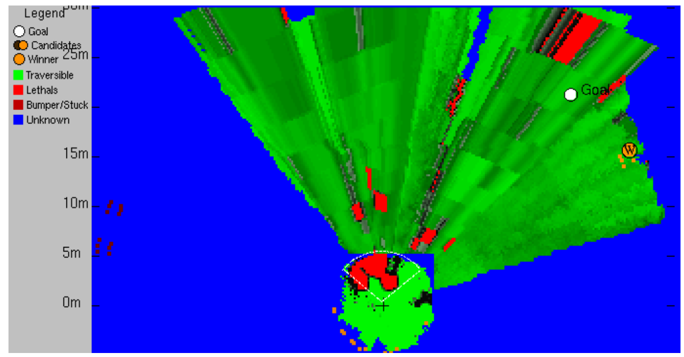

Figure 12. New color scheme applied to Netscale high resolution map.

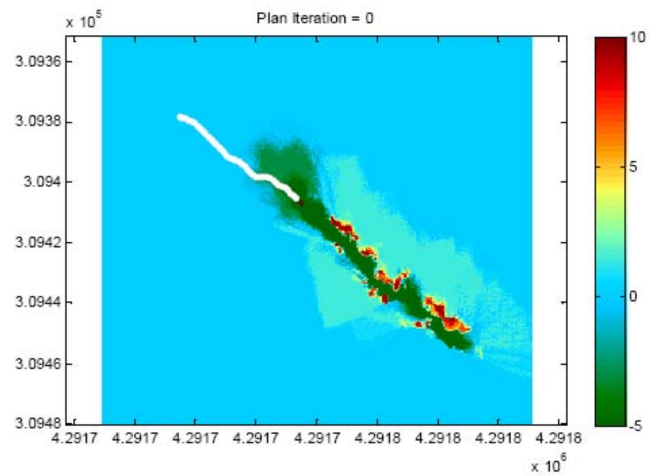

Figure 13. U. Penn's variant of the color scheme.

V Discussion and Conclusions

The color scheme was distributed to the teams in December, 2006. They were expected to try to have it in place by the January, 2007 test, and were mandated to have it in place by the February, 2007 test. While a few of the teams made the deadline, many did not. NIST provided a function to map cost and identification of a pixel or region to color, and now all teams have either adopted the color scheme as is, or, like U. Penn, make use of a variant based on the common color scheme.

Use of the color scheme has had the desired effect. The Government evaluation team has expressed satisfaction in the results. They can more easily understand the OCUs of different teams, although the fact that teams do not have to use a standardized OCU layout makes the benefit smaller than it could have been. There is a lot of resistance in the teams to changing their OCU displays. The displays were developed for debugging and for monitoring the vehicles' progress. Given the very large variation in approaches to the LAGR problems, it would be difficult and very time-consuming to come up with a universal OCU that captured all the possible information to be displayed and also supported the debugging approaches of all the teams.

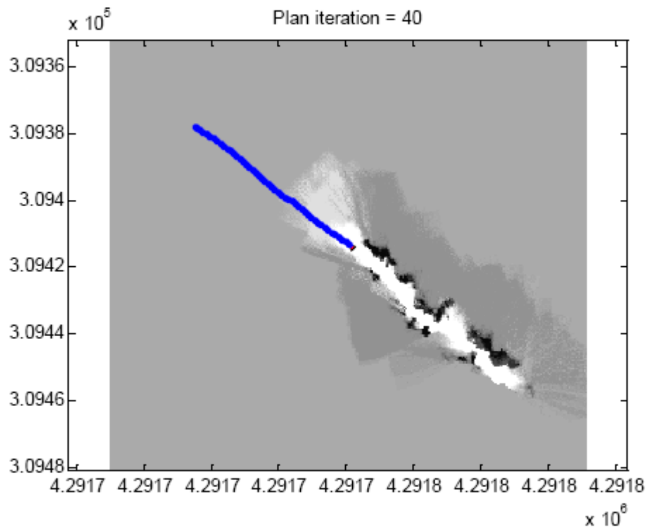

Figure 14. U. Penn's prior color scheme.

It is interesting that there are few, if any, standard color schemes for computer applications. Large vendors, such as Microsoft and Apple, develop and publish in-house styles for their products that cover layout, color, and menus, amongst other things. Typically, colors are chosen from “themes” or groups of compatible colors Thus, all Microsoft Office products have the same look and feel on a single machine. By choosing different themes, different users keep consistency between different tools, but the tools may appear different than those on another machine. This is not a standard, however, although other vendors are encouraged to adopt the color schemes. Other tools for accomplishing similar tasks often use different color schemes to differentiate their products and avoid potential legal issues.

There are many tools to help choose colors that go together [6], but they work primarily for small numbers of colors. For larger numbers, people sometimes use a “natural” color scheme in which a photograph of a natural scene is used to pick the colors. The assumption is that nature is harmonious, so the resulting color scheme will be as well. This approach works well, but doesn’t suit our purposes. We have certain colors that by convention in the program have predefined meanings (e.g., red for obstacles). Some of the colors should stand out from the others, so not all the colors should be harmonious. We also had to accommodate strong feelings on the part of different teams about use of certain colors. All this led us to a consistent, understandable color scheme that breaks many of the rules for picking colors.

It was a useful, if somewhat tedious process to develop the common color scheme. Work needs to be done more broadly to develop standard color schemes for different application areas, such as medical images, geographic information systems, and other complex visual displays that require substantial effort to understand.

Bibliography

[1] D. N. Powell, G. Gilbreath, and M. H. Bruch, "Multi-robot operator control unit," SPIE Proc. 6230: Unmanned Systems Technology VIII, Defense Security Symposium ed Orlando, FL: 2006.

[2] P. K. Pook, J. A. Frazier II, T. Ohm, R. Robert, and G. Whittinghill, "A Testbed for Evaluating Robot Team Interaction under Operator Supervision," Proceedings of UXO Forum '98, http://citeseer.ist.psu.edu/267793.html 1998.

[3] L. D. Jackel, E. Krotkov, M. Perschbacher, J. Pippine, and C. Sullivan, "The DARPA LAGR Program: Goals, Challenges, Methodology, and Phase I Results," *Journal of Field Robotics, Special Issue on Learning in Unstructured Environments*, vol. 23, no. 11/12, pp. 945-973, 2006.

[4] T. Wagner, "Learning Applied to Ground Robots (LAGR)," http://www.darpa.mil/ipto/programs/lagr/index.htm, 2005.

[5] Wikipedia, "Color Theory," http://en.wikipedia.org/wiki/Color_theory, 2007.

[6] P. Lyons and G. Moretti, "Nine Tools for Generating Harmonious Colour Schemes,", The 6th Asian Pacific Conference on Computer Human Interaction (APCHI 2004) ed Rotorua, New Zealand: 2004.