

Design and Implementation of an Industrial 360° Surround View Camera System using IFM Hardware and Multi-Camera Image Processing

Renuka Inamdar

Dept. of Electronics & Telecommunication
Vishwakarma Institute of Information Technology
Pune, India

Ishita Raut

Dept. of Electronics & Telecommunication
Vishwakarma Institute of Information Technology
Pune, India

Vedant Chougale

Dept. of Electronics & Telecommunication
Vishwakarma Institute of Information Technology
Pune, India

Abhishek Deokar

Dept. of Electronics & Telecommunication
Vishwakarma Institute of Information Technology
Pune, India

Abstract - The presence of blind spots in industrial vehicles and mobile machinery creates considerable barriers to effective operator observation of surrounding people, machines and obstacles. The paper outlines the architecture and methods for the implementation of a 360° surround-view technology in industry through the comprehensive application of EcomatDisplay IFM, CODESYS visualization and several streams of video. The system suggested integrates IP cameras working through RTSP in Ethernet with a layer that coordinates the respective camera views and also with a multi-window visualizer. The research suggests to use Modbus TCP protocol to exchange information about the state of the machine with a vehicle controller in order to change layout accordingly to the state of the last in the course of operation (for instance, during the reverse motion). The publication covers such matters as hardware architecture, communication interfaces, video-stream management, dynamic window management and validation technique but suggests that there are no measurements regarding the actual latencies and resource usage defined in the sources and thus requiring verification through experiments.

Keywords - industrial surround view, IFM ecomatDisplay, CODESYS, IP camera, RTSP, Modbus TCP, multi-camera visualization, human-machine interface.

I. INTRODUCTION

Industrial transports such as automated guided vehicles (AGV) and mobile machines typically operate in a challenging environment where the operators do not have direct visibility of the vehicle vicinity. Aspects such as vehicle geometry, cargo, and structure create blind spots. A typical rear-view camera can improve visibility in a single direction, but it cannot create a full picture of the surroundings of the vehicle.

A surround-view system is based on the combination of the camera feeds that surround the vehicle and feed them to an in-cab monitor. For industrial application the design of such system needs to consider such aspects like rugged operating environment, existing hardware for display and processing, as well as camera and machine control signals compatibility.

This paper presents a concept of an industrial surround-view implementation developed using IFM Ecomat Display and CODESYS visualization environment. The design incorporates IP cameras transmitting images via Ethernet with RTSP standard and ensuring multi-window vision, as well as Modbus TCP communication with the vehicle controller. One of the contributions of this study is the common architecture of hardware/software solution and practical validation plan. It should be noted that the developed documentation does not provide the final

- 1) Discuss the communication setup along with the physical devices involved in making a setup for multiple cameras with full 360-degree vision in an industrial setup.
- 2) Explain how multiple camera inputs are transmitted and then displayed on a screen.

- 3) Describe a window layout that would change according to states of the machine it is serving.

- 4) Define all the measurements that need to be taken for checking delays, stability of displays and consumption of resources used in the display setups.

II. RELATED WORK

Surround-view monitoring systems for vehicles have been explored. Cuma, Dükünlü, and Yirik present a 360-degree surround-view camera system and smart driver-behavior recognition technology for electric buses. Hence, the authors provide an example for practical use of multi-camera systems [3]. However, though the operational requirements of the examined case differ from other domains, the authors show the feasibility of various camera usage to gather information about the surroundings in a vehicle.

Another area of application related to industrial automation is automated guided vehicles. The authors Gnapp and others present the use of automatic guided vehicles in the production hall and show the necessary conditions where vehicles can be automated [4]. Research in embedded vision gives more information for developing visual solutions for mobile industrial platforms [5].

The implementation part of the paper should be completed according to IFM and CODESYS documentation [1], [2]. The required information can be found in the mentioned sources, which have to be used when discussing the interfaces, run-time capabilities, and limits of both devices. Unfortunately, the current sources reviewed do not provide enough evidence to prove that the suggested method is better than the previous ones.

III. SYSTEM HARDWARE ARCHITECTURE AND COMMUNICATIONS

A. IFM EcomatDisplay Platform.

The function of the IFM EcomatDisplay is to be the HMI (operator-oriented) as well as to act as a base for visualizing the entire surroundings. The draft data speak about CR1077 and CR1201 families as potential candidates but does not define the exact model used. The final version must therefore mention the model number verified, hardware version, type of video channels supported, and screen resolution. The necessary configuration of the required display can be found from the relevant documentation on IFM system and programming [1]. If the characteristics such as processor, graphics, networking, video decoding of the required display device are checked then the model specifications must be checked. Only when information concerning the enclosure rating and environment is taken from the datasheet.

B. Camera system

The suggested project involves the use of a maximum of four cameras round the vehicle to gain outside views: front, back, left, and right. The number of cameras and their mounting locations should depend on the existing vehicle geometry and needed visibility.

The method of acquiring video fed from IP cameras is based on RTSP streams transiting over Ethernet. RTSP is responsible for the establishment and control of the media session, while the carried actual video can be transmitted using a transport and encoding scheme supported by the camera and receiver.

The draft document states that the realization phase will include the documentation of camera type and model, specification on the form of the RTSP stream, codec used, frame rate, resolution and available network speed.

The draft outlines the discussion of analog cameras as the secondary or legacy option. Their feasibility will depend on whether the display hardware will be able to function with analog video input or whether there is a suitably coded external interface.

C. Machine-State Communication Using Modbus TCP

It is recommended to use Modbus TCP to facilitate communication between the application interface and the vehicle controller or PLC. In doing so, the mapped register area will be able to provide the machine state information like reverse gear status or a request for a specific camera focus. It is crucial to determine which of the two devices will take control over the communication as a client or server. Additionally, it is required to define such elements as register addresses and types of data, frequency of loops, timeout operations, as well as fallback information on the display in case of communication failure.

It is important to note that the machine's state information will be used for displaying purposes only after receiving and validating the data value. All measures associated with safety will be carried out within the framework of the validated control and risk evaluation of the machine. The surround-view display should not take over the role of any certified safety technology.

IV. CODESYS SOFTWARE ARCHITECTURE AND WINDOW MANAGEMENT

A. Video-Stream Handling

The framework captures each of the configured inputs from the cameras, subsequently making the decoded image accessible for use in

a visualization subsystem and instantaneously updating the monitor window. For camera i , the process is as follows:

$S_i(t) = \text{Decode}(\text{Stream}_i(t))$, where $\text{Stream}_i(t)$ is the video data obtained from the i -th camera at time t and $\text{Decode}(\cdot)$ is the decoding operation supported by the system. $S_i(t)$ is the image frame obtained after the above calculations. This formula defines the desired processing path; however, the actual processing route and the presence of hardware acceleration must be verified for the given display and runtime setup. Every camera must have its source configuration and failure state. The visualization should be able to recognize unavailable streams wherever the platform allows it and give a clear information status instead of showing a non-updated picture that could be interpreted as a live image. The number of decoded streams must be verified with the target hardware specification.

B. Window Coordinates and Layout Modes

The application represents each camera window as a rectangle with respect to the position of its upper left and size values:

$$W_i = (x_i, y_i, w_i, h_i)$$

where x_i - horizontal position, y_i - vertical position, w_i - width, h_i - height in pixels. The values of earlier mentioned parameters must be limited to the actual resolution of a system and should be checked during the process of visualization.

With a layout of four views, the whole screen area is divided into four areas to see all four feeds, front, back, left, and right. With the focus layout, the window selected is expanded while the other feeds are minimized. The project should specify and test the size of the windows, transition behavior, and their priority as per CODESYS visualization project.

C. State-Driven View Selection

There will be a simple state machine to manage transitions between overview layout and focused layout. For example, a validated reverse state signal might call for rear camera emphasis. Other triggers should, however, be added only when they are available in the vehicle control interface and their behavior has been defined.

The state machine should define initialization, normal overview, focused view, communication timeout, camera stream failure and recovery behaviors. All transitions should be deterministic, and the application should not switch too quickly between states to avoid being influenced by noisy or inconsistent signals. Any animation or resizing effect should be assessed for readability and for processor load on the target device.

D. Runtime and Configuration

The target application has been deployed on the CODESYS visualization platform. It is recommended to consult the appropriate CODESYS documentation to confirm the visual objects compatible with CODESYS, the runtime being applied, the method of distributing video signal to different receivers, and potential limitations in settings [2]. The final project paper should also indicate which version of CODESYS was used, which target package and libraries were involved, and which realization method was chosen when implementing the project.

V. IMPLEMENTATION AND VALIDATION METHODOLOGY

A. Prototype Configuration

Documentation of the prototype should include a configuration table which captures the exact display model and resolution, camera models, stream options, network topology, CODESYS version, PLC/controller interface, and the software build. As it stands, the source document has incomplete fields such as display model, resolution, stream configuration, and the measured values which need to be updated from the actual test set up on its final release.

B. Functional Verification

In the first step of functional verification each camera should be tested as a standalone unit. The test must ensure that the stream is established, that the video is being displayed in real time, that camera orientation and labels are in place and that the image is visible in the designated window. The tests are to be repeated in the same manner with all streams turned on.

The layout tests are supposed to check the four-camera view, the focused camera, and all intended state transitions. The tests are to assess disconnection of the camera and communication problems with it (e.g. invalid controller data and its recovery after a malfunction)

C. Performance Metrics

The end-to-end video latency is defined as the duration in time between the moment when an event happens and can be visible to a camera and the moment when this event appears on a screen. The technique used for the measurements, as well as the test equipment, the number of test runs, and the statistical method for data processing should be reported. If the latency is measured using a high-speed camera or timestamping techniques, the configuration used and calculated errors must be mentioned.

Other important measures include stream startup time, frame rate, dropped or stalled frames, CPU and RAM utilization, available network bandwidth, responsiveness of a display to layout changes, and the number of streams.

VI. RESULTS AND DISCUSSION

The material states that the initial intended prototype would use an IFM EcomatDisplay and several RTSP cameras, however in the section devoted to numerical results this material gives incomplete results for the display model, resolution, configuration of the stream, end-to-end latency as well as for the operation speed assumptions. This leaves this documentation not sufficient for proving any data-related performance claims associated with the material.

The section in question ought to be revised so that it presents an indication concerning the hardware and software used as well as the results of functional tests and the values connected with performance metrics described in the fifth section. It is necessary at least to specify the number of camera streams tested at same time along with the observed frame rate, end-to-end latency, appearance of dropped frames, and the consumption of resources. Each of the provided values ought to relate to the tested method.

Comparison with another system is supposed to be provided only in case the reference to the original system has been provided in the proper way. The same is true with any statement stating that the latency complies with industrial safety requirement.

VII. LIMITATIONS AND FUTURE WORK

The current version does not finalize the display format, the camera model specifications, the streaming setup and the performance results. The design must be tested on the actual target device, because the availability of the decoding capability and the necessary interfaces and visualization performances is determined by the type of device.

The future work must cover completing the prototype configuration, measuring the end-to-end latency and resource consumption, testing the camera streams in the real lighting and networking conditions, and fault detection and recovery testing. It is also worth mentioning that the camera calibration, alignment, overlapping of the adjacent views, distortion correction and stitching can be included into the scope of the research, however, these possibilities do not present functionalities.

VIII. CONCLUSION

This paper discusses the concept of an industry 360° surround view system from the perspective of the IFM Ecomat Display, CODESYS visualization, the use of video streams from multiple cameras, and

Modbus TCP communication. The architecture shows how the streams coming from the cameras can be processed on a multi-window display and interact with the controller allowing to request a focused view. The source materials are sufficient for illustrating the project architecture and methodology of its validation. However, the received results do not cover all the required data. Thus, before submitting the paper for publication, the detailed information about the hardware used, results of the functional testing, and performance evaluations would have to be provided.

REFERENCES

- [1] IFM Electronic, "ecomatDisplay System & Programming Manual: CODESYS V3.5 System Description," IFM Technical Documentation, 2024.
- [2] CODESYS GmbH, "CODESYS Visualization and TargetVisu Architecture Guide," CODESYS Technical Manual, 2023.
- [3] M. U. Cuma, Ç. Dükünlü, and E. Yirik, "Smart Driver Behavior Recognition and 360-Degree Surround-View Camera for Electric Buses," *Electronics*, vol. 12, no. 13, Art. no. 2979, 2023.
- [4] J. Gnap, I. Kubasáková, J. Kubáňová, and D. Pauer, "Application of Technological Procedure Automated Guided Vehicles in the Production Hall of a Company Due to Increasing the Automation," *Applied Sciences*, vol. 14, no. 1, 2024.
- [5] S. S. A. Zaidi et al., "Embedded vision systems for industrial automated guided vehicles: A review," *IEEE Access*, vol. 9, pp. 102140–102158, 2021.