ANSYS SCADE Vision
Release 2019 R3

Purpose of this Document

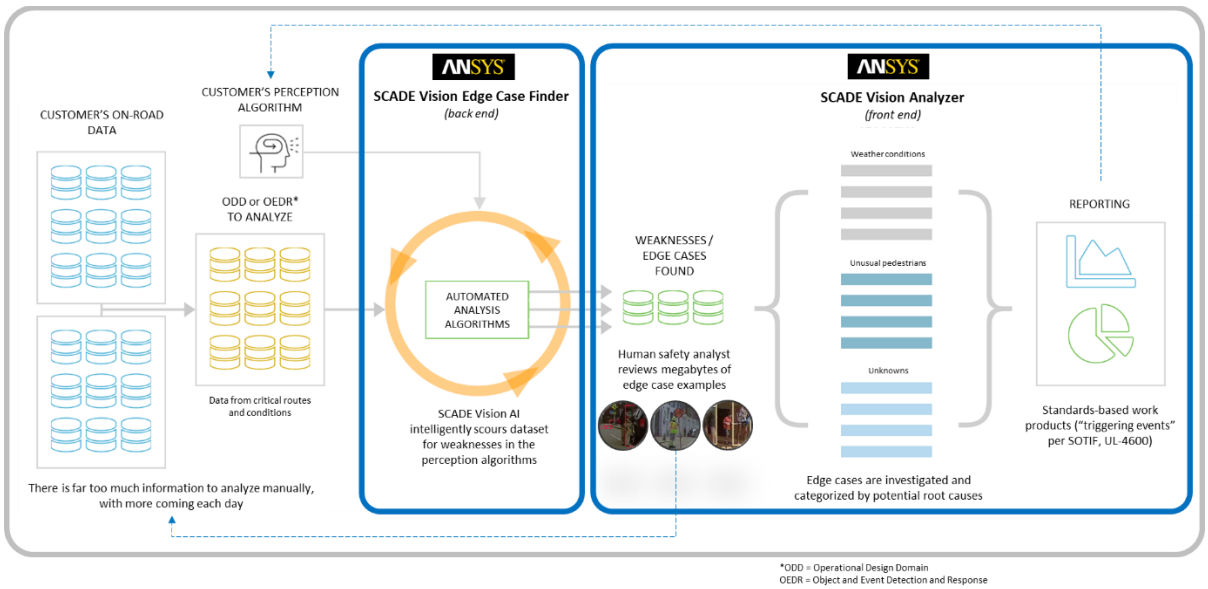
The purpose of this document is to provide a description of the modules that are part of the ANSYS SCADE Vision product line.

Table of Contents

1 ANSYS SCADE Vision

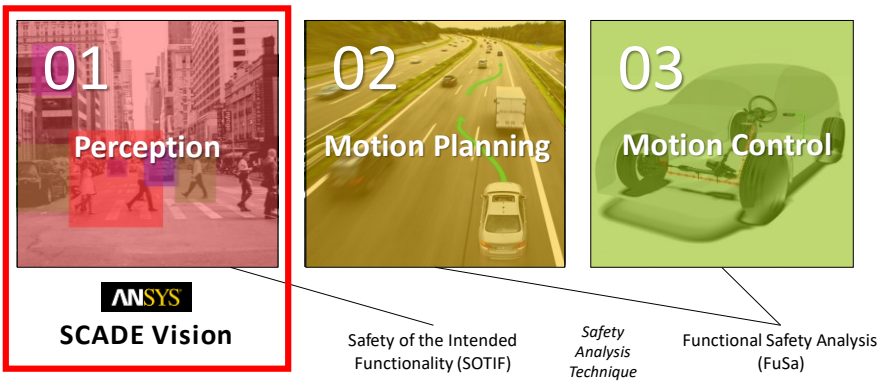
This document provides a description of the modules that are part of ANSYS SCADE Vision family of products and solutions, powered by Hologram, which is composed of an automated testing and analysis engine, the SCADE Vision Edge Case Finder, and a web-based UI to aid human analysts, the SCADE Vision Analyzer.

The positioning of these modules within an overall workflow is shown in the figure below.



SCADE Vision is a powerful platform used to efficiently identify triggering events in visual perception algorithms used for ADAS and AV products. From ISO 21448 PAS, Safety of the Intended Function (SOTIF), triggering events are "*specific conditions of a driving scenario that serve as an initiator for a subsequent system reaction possibly leading to a hazardous event*".

SCADE Vision speeds up the discovery of weaknesses in your AV embedded perception software that may be tied to edge cases, and helps identify their root causes – also called triggering events – by automatically applying augmentations to the data coming from your sensors (such as raw video) to identify fragility in your AI-based embedded perception software. Without SCADE Vision, triggering event analysis depends entirely on manual labeling of video data, which is extremely time consuming and expensive.



SCADE Vision is intended for safety analysts, neural network developers and testers and anyone else who works with machine learning-based perception systems.

ANSYS SCADE Vision is composed of the modules described in the following paragraphs.

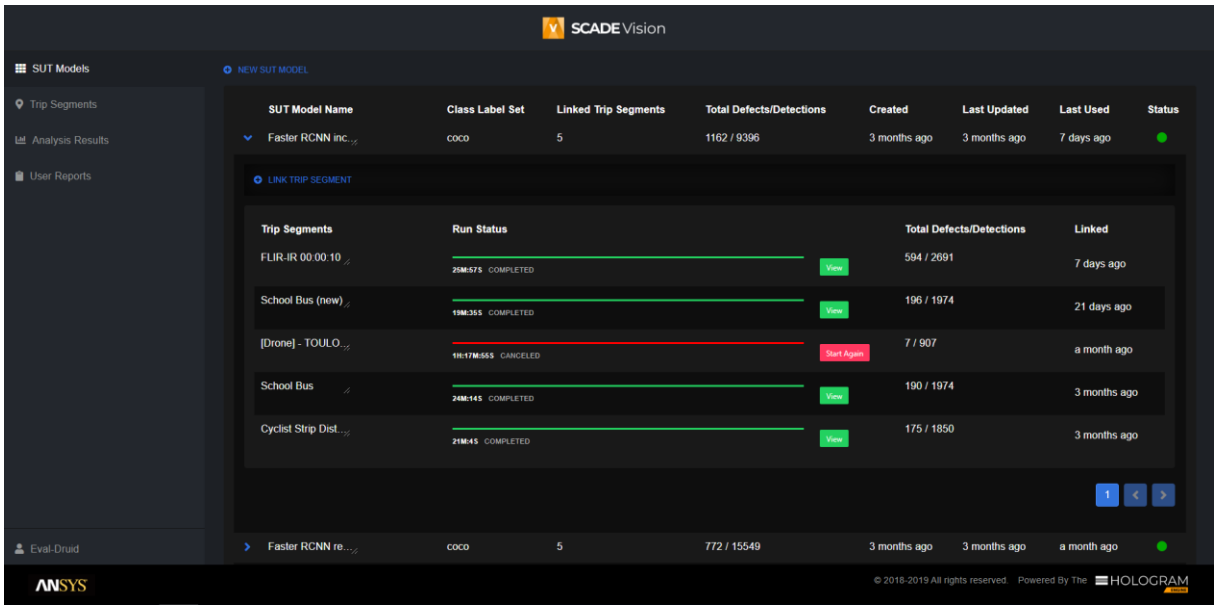
1.1 *ANSYS SCADE Vision Analyzer*

Purpose

The SCADE Vision Analyzer is a web-based User Interface (UI) which provides users with the ability to select the perception software under test and sensor raw data files, launch the analysis, browse probable defects pinpointed by the SCADE Vision Edge Case Finder in the perception software, categorize these defects into proposed triggering events, and to produce reports for safety and development teams.

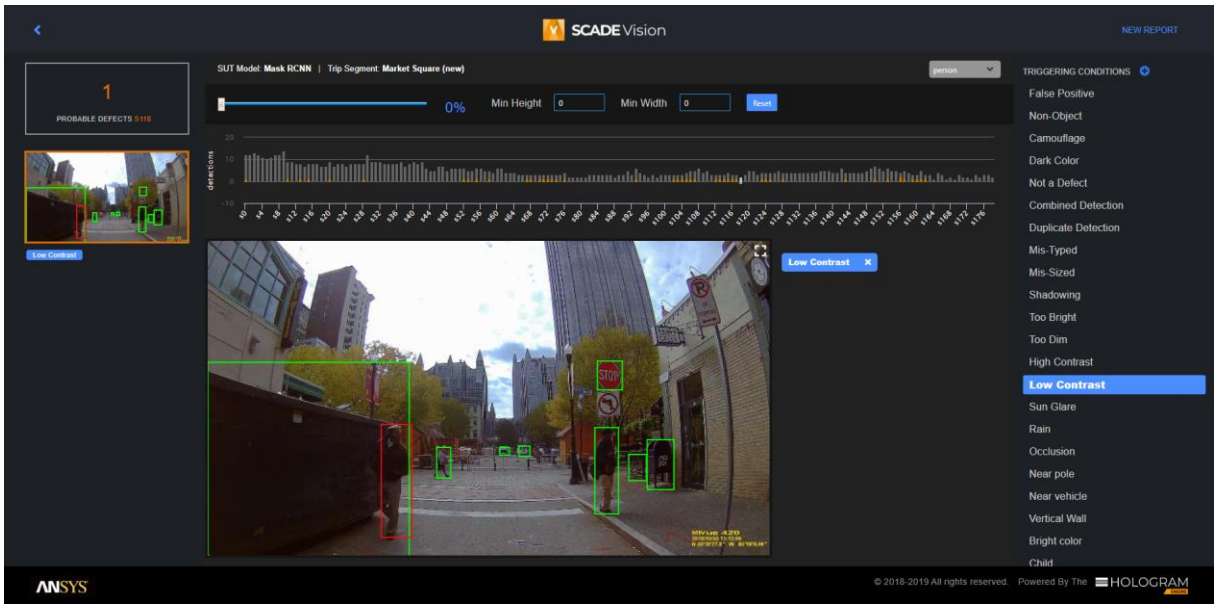
Capabilities

Connecting the System-Under-Test. The Analyzer is the portal through which systems under test (SUTs), the perception software, are uploaded into the SCADE Vision cluster. These SUTs are convolutional neural networks (CNNs) for object detection. The Analyzer supports uploading TensorFlow models, one of the most popular deep learning frameworks in which CNNs are developed, an open source framework used by many ADAS and AV developers. The Analyzer currently supports text labels for each SUT version, so that test results can be traced to that version.



Submitting video data from trips. The Analyzer supports uploading of video files into the SCADE Vision cluster. These video files must be in MPEG4 format. The Analyzer supports a concept of a "trip" – a labeled route or scenario reflected in the video. Test results are traced to trip labels. The Analyzer user can submit trips to the Edge Case Finder, which processes them in the background, allowing the user to perform triggering event analysis while processing proceeds.

Discovering triggering events. Once results have been posted to the SCADE Vision database, the Analyzer discovery page can visualize them. The discovery page presents the user with a time series of scenes – in this case, frames of video. The user selects which object type to analyze (e.g., pedestrian, car, etc.) and for each frame, the Analyzer displays (1) the number of relevant objects detected and (2) the number of probable defects that SCADE Vision pinpoints in that scene. The user clicks on scenes with defects to see thumbnails of each frame with potential defects identified by SCADE Vision.

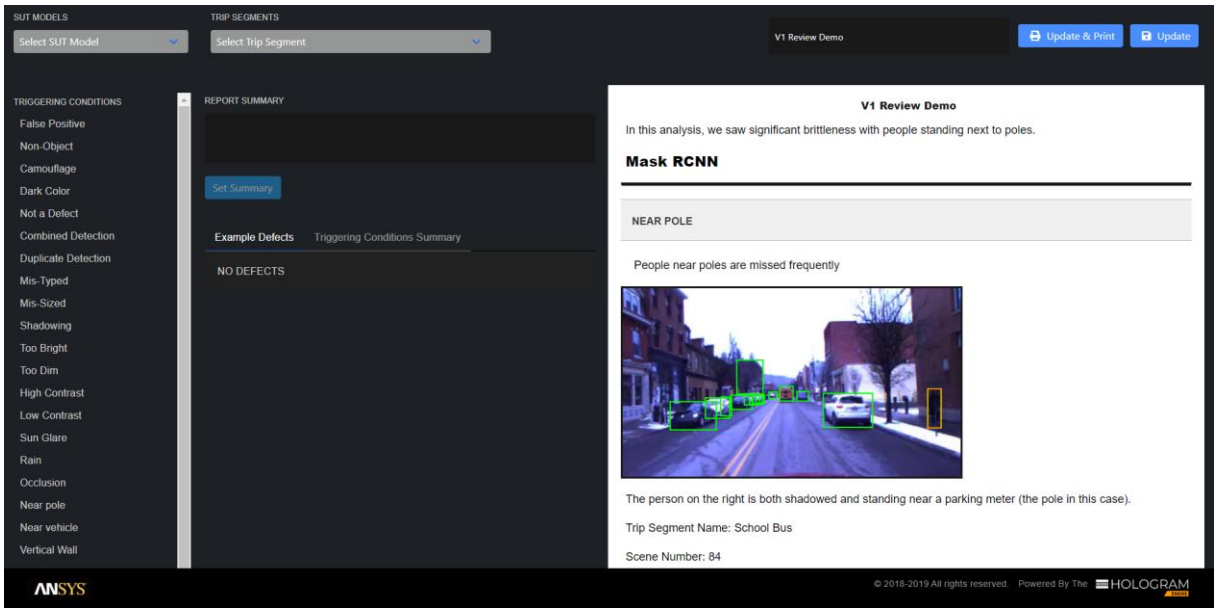


The user's task in the discovery page is to tag all relevant defects with potential triggering events. The Analyzer suggests a set of standard tags which trace to documented sources: the SOTIF standard, CV-HAZIP, and so on.

Triggering events are diverse: weather conditions (snow, rain, wildfire), lighting conditions (glare, night, high beams), infrastructure (fences, reflective surfaces, statues), types of road users (wheelchairs, people in costumes), or simply an incomplete training of the AV machine learning systems. Standardized tags are useful for consistent triggering event analysis across a team of SCADE Vision users. However, part of the challenge of triggering event analysis is tackling "unknown unknowns", which implies that any concrete set of tags may be insufficient for newly discovered kinds of SUT limitations. Therefore, the Analyzer also supports customized tags that can be edited to account for newly-discovered effects.

Generating safety reports. Once all relevant defects have been tagged, then the user can move to the reporting phase. The reporting is structured around a summary of triggering event tags that have been identified. The purpose of a report is for the SOTIF analyst to give guidance to system engineering and software development teams about what mitigations should be implemented, per SOTIF. To help explain the triggering events proposed, the user can select examples for the report. For each triggering event identified, the user can then suggest a mitigation strategy. These strategies can trace to those listed in SOTIF. Additional text can be entered to help guide mitigations.

Automatically generated reports help you to structure and communicate the results of the safety analysis with members of the AV perception software development team and other interested parties, in a virtual feedback loop.



Alignment with the SOTIF (Safety Of The Intended Functionality) standard

The features of the Analyzer are aligned with tasks described in SOTIF:

Identification and evaluation of triggering events. The Analyzer establishes a systematic method to perform an analysis of triggering events. It aims to identify weaknesses in perception algorithms and the related scenarios that could lead to an identified hazard. By testing perception in real-world, on-road conditions, this analysis can consider categories such as environment and location, road infrastructure, urban infrastructure, highway infrastructure, algorithm limitations, mechanical disturbances, interference, glare, poor-quality reflections, accuracy, range, and more.

Functional modifications to reduce SOTIF related risks. The Analyzer report generation features play an important role in planning functional modifications to reduce the risks posed by triggering events due.

Definition of the verification and validation strategy. SCADE Vision can be used to support V&V methods such as error guessing based on knowledge or experience; analysis of common limit conditions, sequences, and sources of dependent failures; analysis of environmental conditions; analysis of sensor design and known limitations; and analysis of triggering events.

Verification of the SOTIF. The examples found by the Analyzer can be used to inject inputs into the perception algorithm under test that trigger the potentially hazardous behavior.

Validation of the SOTIF. The Analyzer is primarily focused on enabling exploratory validation use cases; put another way, it is designed to help find "unknown unknowns" efficiently. This is a very much iterative and ongoing process during development. It traces to methods for evaluating residual risk such as in-the-loop testing on randomized test cases (derived from a technical analysis and by error guessing), randomized input tests; long-term vehicle tests (though SCADE Vision does so in an open-loop fashion only on the recorded sensor data); tests of corner cases; and analysis of worst-case scenarios.

1.2 ANSYS SCADE Vision Edge Case Finder

Purpose & Capabilities

The SCADE Vision Edge Case Finder enables accelerating the testing of the customer AI-based perception software under test (SUT), usually a convolutional neural network (CNN). Testing consists of automatically running the SUT inference algorithm twice against each raw input video, with the following phases:

- **Trip data replay.** The Edge Case Finder replays the baseline, unmodified video, from trips selected and submitted by the Analyzer users. It decodes this video, submits each frame to the SUT, and stores the SUT outputs in the results database.
- **Data modification.** For each frame that includes a potential objects of interest (e.g. pedestrians, cars), the Edge Case Finder modifies that frame based on an augmentation that is effective for defect detection in the perception software (such as Gaussian noise). The Edge Case Finder submits this modified frame to the SUT and stores the SUT outputs in the results database.
- **Run defect analysis algorithms.** The Edge Case Finder analyzes the SUT outputs stored in the results database using several potential defect analysis algorithms (including weak detection and false negative prediction). The output of these algorithms are pointers to scenes in the trip segments in which the SUT exhibits defects. These pointers are also stored in the results database for analysis in the Analyzer.

SUTs and trip data are input data for the Edge Case Finder, submitted through the Analyzer, for which the scalability in execution is architected into the Edge Case Finder.

SCADE Vision does not require labeled data to support AV perception software testing; instead, it searches through raw sensor data recorded by the autonomous vehicles, or video synthetized from car driving simulator (such as ANSYS VRXPERIENCE).

Deployment

The Edge Case Finder is a back-end application. Both the Analyzer and the Edge Case Finder are deployed using Kubernetes, typically to GPU-enabled EC2 nodes on Amazon Web Services (AWS) EC2 nodes. The use of Kubernetes supports running the Edge Case Finder on-premises, or on other cloud platforms, instead of on AWS.