CV-HAZOP: Introducing Test Data Validation for Computer Vision

Oliver Zendel, Markus Murschitz, Martin Humenberger, Wolfgang Herzner
AIT Austrian Institute of Technology
Donau-City-Strasse 1, 1220 Vienna, Austria
{oliver.zendel,markus.murschitz.fl,martin.humenberger,wolfgang.herzner}@ait.ac.at

Abstract

Test data plays an important role in computer vision (CV) but is plagued by two questions: Which situations should be covered by the test data and have we tested enough to reach a conclusion? In this paper we propose a new solution answering these questions using a standard procedure devised by the safety community to validate complex systems: The Hazard and Operability Analysis (HAZOP). It is designed to systematically search and identify difficult, performance-decreasing situations and aspects. We introduce a generic CV model that creates the basis for the hazard analysis and, for the first time, apply an extensive HAZOP to the CV domain. The result is a publicly available checklist with more than 900 identified individual hazards. This checklist can be used to evaluate existing test datasets by quantifying the amount of covered hazards. We evaluate our approach by first analyzing and annotating the popular stereo vision test datasets Middlebury and KITTI. Second, we compare the performance of six popular stereo matching algorithms at the identified hazards from our checklist with their average performance and show, as expected, a clear negative influence of the hazards. The presented approach is a useful tool to evaluate and improve test datasets and creates a common basis for future dataset designs.

1. Introduction

Many safety-critical systems depend on CV technologies to navigate or manipulate their environment and require a strong safety assessment due to the evident risk to human lives [24]. Unfortunately, people working in the field of CV will often notice that algorithms scoring high in public benchmarks [32] perform rather poor in real world scenarios. The reason is that those limited benchmarks are applied to open world problems. While every new proposed algorithm is evaluated based on these benchmark datasets, the datasets themselves rarely have to undergo independent evaluation.

This work presents a new way to facilitate this safety assessment process and goes beyond the basic scope of benchmarking: applying a proven method used by the safety community for the first time to the CV domain. It provides an independent measure that counts the challenges in a dataset that are testing the robustness of CV algorithms.

The typical software quality assurance process uses two steps to provide objective evidence that a given system fulfills its requirements: verification and validation [16]. Verification checks whether or not the specification was implemented correctly (*e.g.* no bugs) [2]. Validation addresses the question whether or not the algorithm meets the original requirements, *e.g.*, is robust enough under difficult circumstances. Validation is performed using test datasets as inputs and comparing the algorithm's output against the expected results (*ground truth, GT*).

While general methods for verification can be applied to CV algorithms, the validation part is rather specific. A big problem when validating CV algorithms is the enormous amount of possible input datasets, *i.e.* test images (*e.g.* for 640×480 8bit image inputs there are $256^{640*480} \approx 10^{739811}$ different test images). An effective way to overcome this problem is to find equivalence classes and to test the system with a representative of each class. However, the definition of equivalence classes for CV is tough: How can we describe in mathematical terms, say, all possible images which show a tree or not a car? This leads to two main challenges for CV validation:

1. What should be part of the test dataset to ensure that the required level of robustness is achieved?
2. How can redundancies be avoided to allow the identification of problematic flaws (wastes time and creates a bias towards repeated elements)?

Traditional benchmarking tries to characterize performance of multiple implementations using fixed datasets to create a ranking based on this data. Contrary, validation tries to show that the algorithm can reliably solve the task at hand, even under difficult conditions. Although both use application specific datasets, their goals are not the same and benchmarking sets are not suited for validation.

The intent of validation is to find shortcomings and poor performance by using "difficult" test cases [34]. The main challenge for validation in CV is a definition of elements and specific relations which are known to be "difficult" for CV algorithms (comparable to optical illusions for humans). In this paper, the term *visual hazard* will refer to such elements and specific relations. By creating an exhaustive checklist of these visual hazards we can answer the questions above:

1. Ensure completeness of test datasets by including all relevant hazards from the list.
2. Reduce redundancies by excluding test data that only contains hazards that are already identified.

Our main contributions presented in this paper are:

- application of the HAZOP risk assessment method to the CV domain (Sec. 3),
- creation of a generic CV system model useful for risk analysis (Sec. 3.1),
- a publicly available hazard checklist (Sec. 3.6) and a guideline for using this checklist as a tool to measure hazard coverage of test datasets (Sec. 4).

To evaluate the feasibility of the approach, the guideline is applied to three stereo vision test datasets: KITTI, Middlebury 2006 and Middlebury 2014. The impact of identified hazards on the output of multiple stereo vision algorithms is compared in Section 6.

2. Related Work

Bowyer *et al*. [6] analyze the problems coming along with validating CV systems and propose that the use of sophisticated mathematics goes hand in hand with specific assumptions about the application. If those assumptions are not correct, the actual output in real world scenarios will deviate from the expected output.

A very popular CV evaluation platform is dedicated to stereo matching, the Middlebury stereo database. Scharstein *et al*. [32] developed an online evaluation platform which provides stereo datasets consisting of the image pair and the corresponding GT data. The datasets show indoor scenes and are created with a structured light approach [33]. Recently, an updated and enhanced version was presented which includes more challenging datasets as well as a new evaluation method [31]. To provide a similar evaluation platform for road scenes, the KITTI database was introduced by Geiger *et al*. [10]. A general overview of CV performance evaluation can be found in Thacker *et al*. [39]. They summarize and categorize the currently used techniques for performance validation of algorithms in different subfields of CV. Some examples are shown in the following: Bowyer *et al*. [5] present a work for edge detection evaluation based on Receiver Operator Characteristics (ROC) curves for 11 different edge detectors. Min *et al*. [26] describe an automatic evaluation framework for range image segmentation which can be generalized to the broader field of region segmentation algorithms. Strecha *et al*. [37] present a multi-view stereo evaluation dataset that allows to evaluate pose estimation and multi-view stereo with and without camera calibration. They additionally incorporate GT (LIDAR-based) quality in their method to enable fair comparisons between benchmark results. Kondermann *et al*. [20] discus the effect of GT quality on evaluation and propose a method to add error bars to disparity GT.

Ponce *et al*. [29] analyze existing image classification test datasets and report a strong database bias. Typical poses and orientations as well as lack of clutter create an unbalanced training set for a classifier that should work robustly in real world applications. Pinto *et al*. [28] demonstrate by a neuronal net, used for object recognition, that the currently used test datasets are significantly biased. Torralba and Efros [40] successfully train image classifiers to identify the test dataset itself (not its content), thus, showing the strong bias each individual dataset contains. Current test datasets neither provide clear information about which challenges are covered nor which issues remain uncovered. Our approach can fill both gaps: By assigning a reference-table entry to each challenging hazard, we create a checklist applicable to any dataset. To the best knowledge of the authors there is no published work considering the vision system as a whole, which identifies risks on such a generic level.

2.1. Robustness

In the safety context, robustness is about the safe handling of abnormal situations or input data in general. In CV robustness is mostly considered to refer to a statistic definition. Popular methods like RANSAC (Random Sample Consensus) [9], M-Estimators [14] in general, or specific noise modeling techniques arose of the need to use systems in "real world applications". However, these methods are not necessarily tackling the problem of robustness by the safety definition. They do not assess the system's response to highly unexpected data. In the safety context, the analysis of the vulnerabilities with respect to robustness are based on so-called Fault Injection Methods [12] (*e.g*. Fuzztesting [38]).

2.2. Risk Analysis

Risk-oriented analysis methods are a subset of validation and verification methods. Basically, all technical risk analysis methods have the goal to assess one or several risk related quality attributes (*e.g*. safety or reliability) of systems, components or even processes with respect to causes and consequences. In addition, they try to identify existing risk reduction measures and to propose additional ones if necessary. Since the topic of risk identification became relevant, first in chemical industries around 1980, it was also applied to software (see Fenelon *et al*. [8] and Goseva-

Popstojanova *et al*. [11] for UML models). The most commonly used methods are:

- HAZOP [7, 19] - Hazard and Operability Analysis,
- FME(C)A [1] - Failure Modes, Effects, (and Criticality) Analysis,
- FTA [42, 22] - Fault Tree Analysis.

All risk analysis methods define systematic processes for identifying potential risks. The first step in a HAZOP is to identify the essential components of the system to be analyzed. The parameters for each component, which define its behavior, have to be identified. These parameters often describe the I/O of the component. A set of predefined guide words which describe deviations are applied to the parameters (*e.g*. "less" or "other than") and the resulting combinations are interpreted by experts in order to identify possible consequences (potential hazards) and counteractions. While FME(C)A also starts with identifying the system's components and their operating modes, it then identifies the potential failure modes of the individual components. Further steps are comparable to the HAZOP.

FTA starts with a hazardous "top event" as root of the fault tree. Recursively, to each current bottom event, leaves are added in Boolean combination that can lead to that event until elementary events are encountered (*e.g*. "own car hits the front car" if "speed too high" and "braking insufficient").

3. CV-HAZOP

The identification and collection of CV hazards should follow a systematic manner and the results should be applicable to many CV solutions. To create an accepted tool for the validation of CV systems, the process has to be in line with well-established practices from the risk and safety assessment community. The most generic method HAZOP [19] is chosen over FME(C)A and FTA because it is feasible for systems for which little initial knowledge is available. In addition, the concept of guide words adds a strong source of inspiration that all other concepts are missing.

The following Sections address the main steps of a HAZOP:

1. Model the system.
2. Partition the model into subcomponents, called locations.
3. Find appropriate parameters for each location which describe its configuration.
4. Define useful guide words.
5. Assign meanings for each guide word / parameter combination and derive consequences as well as hazards from each meaning.

3.1. Generic Model

The first step of any HAZOP is deriving a model of the system that should be investigated. In case of this HAZOP,

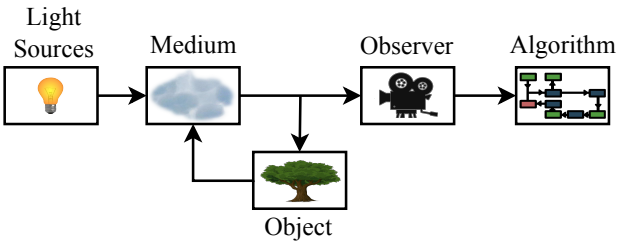

Figure 1. Information flow within the generic model

the generic CV algorithm has to be modeled together with the observable world (its application). Marr [23] proposes a model for vision and image perception from the human perception perspective. Aloimonos and Shulman [3] extended it by the important concepts of stability and robustness. We propose a novel model which is entirely based on the idea of information flow: The common goal of all CV algorithms is the extraction of information from image data. Therefore "information" is chosen to be the central aspect handled by the system. It should be noted, that "information" is used in the context "Information is data which has been assigned a meaning." [41] rather than in a strict mathematical sense [36]. In this context, hazards are all circumstances and relations that cause a loss of information. Even though hazards ultimately propagate to manifest themselves in the output of the algorithm, an effective way to find a feasible list of hazards is to look at the entire system and attribute the hazard to the location where it first occurred (*e.g*. unexpected scene configuration or sensor errors). Multiple inputs from different disciplines are used to create the system model:

Information Theory: Communication can be abstracted according to information theory [36] as information flow from the transmitter at the source with the addition of noise to the receiver at the destination.

Sampling Theorem: Sampling is a key process in the course of transforming reality into discrete data. Artifacts that can be caused by this process, according to Nyquist [27] and Shannon [35], will result in a loss of information.

Rendering Equation: The rendering equation [18] is a formal description of the process of simulating the output of a virtual camera within a virtual environment. The different parts of the standard rendering equation amount to the different influences that arise when projecting a scenery light distribution into a virtual camera.

The entire flow of information is modeled as follows:

1. Initially all necessary data of an observed scene exists. In addition, every necessary interpretation of it is given to derive full and precise information for solving the problem at hand.
2. All information about the observed scene available to a CV component can only be provided by the electromagnetic spectrum (simply referred to as "light" in

Guide Word	Meaning	Example
Basic		
No	No information can be derived	No light at all is reflected by a surface
More	Quantitative increase (of parameter) above expected level	Spectrum has a higher average frequency than expected
Less	Quantitative decrease below expected level	Medium is thinner than expected
As well as	Qualitative increase (additional situational element)	Two lights shine on the same object
Part of	Qualitative decrease (only part of the situational element)	Part of an object is occluded by another object
Reverse	Logical opposite of the design intention occurs	Light source casts a shadow instead of providing light
Other than	Complete substitution - another situation encountered	Light source emits a different light texture
Additional - Spatial		
Where else	"Other than" for position / direction related aspects	Light reaches the sensor from an unexpected direction
Spatial periodic	Parameter causes a spatially regular effect	A light source projects a repeating pattern
Spatial aperiodic	Parameter causes a spatially irregular effect	The texture on object shows a stochastic pattern
Close / Remote	Effects caused when s.t. is close to / remote of s.t. else	Objects at large distance appear too small
In front of / Behind	Effects caused by relative positions to other objects	One object completely occludes another object
Additional - Temporal		
Early / Late	Deviation from temporal schedule	Camera iris opens too early
Before / After	A step is affected out of sequence, relative to other events	Flash is triggered after exposure of camera terminated
Faster / Slower	A step is not done with the right timing	Object moves faster than expected
Temporal periodic	Parameter causes a temporally regular effect	Light flickers periodically with 50Hz
Temporal aperiodic	Parameter causes a temporally irregular effect	Intensity of light source has stochastic breakdowns

Table 1. Guide Words used in the CV-HAZOP

this paper) received by the "observer" (*i.e.* the sensor / camera) from any point in the scene. Hence, the information in the scene has to be converted into light in some way.

3. At the same time, any generation of light and interaction of light with the scene distorts and reduces this information.
4. The sensing process, *i.e.* the transformation of received light into digital data, further reduces and distorts the information carried by the received light.
5. Finally, the processing of this data by the CV algorithm may also loose or distort information (through rounding errors, integration etc.).

In essence, two information carriers are distinguished: light outside of the system under test (SUT) and digital data within the SUT. At each transition, this information can be distorted (*e.g.* by reduction, erasure, transformation, and blending).

3.2. Locations

The system model is now partitioned into specific locations (*i.e.* subsystems) of the overall system. Light sources that provide illumination start the process flow (illustrated in Fig. 1). The light traverses through a medium until it either reaches the observer or interacts with objects. This subprocess is recursive and multiple interactions of light with multiple objects are possible. The observer is a combination of optical systems, the sensor, and data pre-processing. Here the light information is converted into digital data as input to a CV algorithm. The CV algorithm processes the data to extract information from it.

Each entity (box in Fig. 1) represents a location for the HAZOP. The recursive pattern (loop) results in an additional location called "Objects" for aspects arising from the interactions between multiple objects. For convenience, the observer is represented by two components: "Observer - Optomechanics" and "Observer - Electronics".

Parameter	Meaning
Transparency	Dimming factor per wavelength and distance unit
Spectrum	Color, *i.e.* richness of medium with respect to absorption spectrum (isotropic or anisotropic)
Texture	Generated by density fluctuations and at surfaces (*e.g.* water waves)
Wave properties	Polarization, coherence
Particles	Influences and effects of the particles that make up the medium

Table 2. Parameters used in the location Medium

3.3. Parameters

Each location is characterized by parameters. They refer to physical and operational aspects that describe the configuration of the subcomponent. The set of parameters chosen for a single location during the HAZOP should be adequate for its characterization. Table 2 shows the parameters chosen for the location "Medium" as an example[1].

3.4. Guide Words

A guide word is a short expression that shall help to trigger the imagination of a deviation from the design / process intent. Number and extent of guide words must be selected to ensure a broad view on the topic. Nevertheless, their number is proportional to the time needed for performing the HAZOP, so avoiding redundant guide words is essential.

[1] See supplemental material for all parameters

The used guide words and their interpretation in the context of CV are listed in Table 1. The first seven "basic" guide words are standard guide words used in every HAZOP and the remainder are adaptations and additions.

3.5. Execution

The actual execution of the HAZOP is the systematic investigation of each combination of guide words and parameters at every location in the system. It is performed redundantly by multiple contributors. Afterwards, the results are compared and discussed to increase quality and completeness. Each HAZOP contributor assigns at least one meaning per combination. In addition, for each meaning found the contributors investigate the direct consequences of this deviation on the system and subsequently into which hazards this can be translated. One meaning can result in multiple consequences at different levels and they are per se not considered harmful or helpful. The hazards, on the other hand, amount to actual decreases in the total system's performance or quality. Combinations that result in meaningful interpretations by any contributor are considered to be "meaningful" entries while combinations without a single interpretation are considered to be "meaningless". The execution of the CV-HAZOP, including various meetings and discussions by the contributors (with expertise in testing, analysis, and CV), took one year. Each location is covered by at least three of the authors. The additional experts are mentioned in the acknowledgments. The 52 parameters from all seven locations, combined with the 17 guide words, result in 884 combinations. Each combination can have multiple meanings assigned to it. Finally, 947 unique and meaningful entries have been produced. Table 3 shows an excerpt of entries from the final HAZOP[2]. The entries in the list can include multiple meanings for each parameter as well as multiple consequences and hazards per meaning. The whole resulting dataset of the CV-HAZOP is publicly available at www.vitro-testing.com.

3.6. Resulting List

In total, 947 entries are considered meaningful by the experts. A detailed analysis of the resulting amount of meaningful entries achieved for each guide word / parameter combination is shown in Fig. 3. One goal is to maximize the meaningful entries - and the graphic shows reasonably high entries for most of the basic guide words (see Tab. 1). Lower valued entries in the matrix can be explained as well: The concepts of the spatial aspects "Close" and "Remote" are simply not applicable to the properties of the electronic part of the observer (Obs. Electronics) and the concept of space in general is not applicable to a number of parameters at various locations. This also counts for the temporal guide words which are not fitting very well to the Optomechanical and Medium locations. Nevertheless, even here the usage of guide word / parameter combinations inspire the analysts to find interpretations which would have been hard to find otherwise.

[2]See supplemental material for more entries

4. Application

The remainder of this paper focuses on the application of the checklist as an evaluation tool for existing datasets. The creation of new test datasets is the logical next step as our checklist can also be used to optimize completeness in terms of robustness, but this is beyond the scope of this paper. Initially, the evaluators have to clarify the intent and domain of the specific task at hand. This specification creates the conceptual borders that allow the following analysis to filter the hazards. The intent includes a description of the goals, the domain defines the conditions, and the environment under which any algorithm performing the task should work robustly. With the intent and domain specified, the evaluators can now check each entry of the CV-HAZOP list to see if that entry applies to the task at hand. Often it is useful to reformulate the generic hazard entry for the specific algorithm to increase readability. In the following a process outline is given:

1. Check if the preconditions defined by the column *Meaning* and the according *Consequences* apply.
2. Check if the *Hazard* itself applies to the specific task.
3. If the *Hazard* in the list is too generic to be feasible, specify the hazard for the specific task.

Previous evaluations for comparable tasks can be used as templates to speed up this process and to reduce the effort compared to evaluating the whole generic list. Specialized hazards can be added to the checklist so that they can be used directly in future evaluations. With the reduced list of possible hazards, the evaluators are able to go through test datasets and mark the occurrence of a hazard. Usually a simple classification per test case is enough. Individual pixel-based annotations can also be used to indicate the location of specific hazards in test images (see Sec. 5). After this process, the missing hazards are known and quantifiable (*e.g.* 70% of all relevant hazards are tested using this test dataset). This is a measure of completeness which can be used to compare datasets. Even more important: If a hazard cannot be found in the test data, the CV-HAZOP entry states an informal specification for creating a new test case to complement the test dataset. The extensiveness of the checklist allows a thorough and systematic creation of new test datasets without unnecessary clutter.

5. Example

As proof of concept, the authors apply the described process to a specific task. For simplicity, we chose canonical stereo vision: The intent of the algorithm is the calcula-

Location/Parameter	Guide Word	Meaning	Consequences	Hazards
Light source / Intensity	More	Light source shines stronger than expected	Too much light in scene	Overexposure of lit objects
Object / Reflectance	As well as	Obj. has both shiny and dull surface	Diffuse reflection with highlight/glare	Object recognition distorted by glares
Object / Texture	No	Object has no texture	Object appears uniform	No reliable correspondences can be found
Objects / Close	Reflectance	Reflecting Obj. is closer to Observer than expected	Reflections are larger than expected	Mirrored scene taken for real
Objects / Positions	Spatial periodic	Objects are located regularly	Same kind of objects appear in a geometrically regular pattern	Individual objects are confused
Optomechanics / Aperture	Where else	Inter-lens reflections project outline of aperture	Ghosting appears in the image	Aperture projection is misinterpreted as an object
Electronics / Exposure	Less	Shorter exposure time than expected	Less light captured by sensor	Details uncorrelated due to underexposure

Table 3. Excerpt from CV-HAZOP Entries (simplified)

Figure 2. Examples for each entry in Table 3 taken from the datasets described in Table 4

tion of a dense disparity image (correspondence between the pixels of the image pair) with a fixed epipolar, two camera setup. To further simplify the analysis, we only use greyscale information and assume that the cameras are perfectly synchronous (exposure starts and stops at the same instants), and omit the use of any history information so that many time artifacts can be disregarded. Note that this evaluation is not designed to compare stereo vision algorithms themselves but to be a clear proof of concept for the usefulness of the CV-HAZOP list. The simplifications in domain / intent analysis and algorithm evaluation were performed to reduce complexity / workload and should be re-engineered for a specific stereo vision evaluation. The domains of the algorithm are indoor rooms or outdoor road scenarios. Problematic conditions like snow, fog, and rain are included.

First, six experts in the field of CV (some had experience with the CV-HAZOP list, others were new to the concept) analyzed the initial 947 entries and identified those applying to the stereo vision use case. During this step, 552 entries are deemed to be not applicable and 106 entries are non-determinable (not verifiable by only surveying the existing test data; more background knowledge needed). The remaining 289 entries are deemed to be relevant for stereo vision. About 20% of the hazard formulations are further specified to simplify the following annotation work while the rest is already specific enough. For each identified hazard, the experts analyzed three test datasets commonly used for stereo vision evaluation (see Tab. 4).

The hazard entries are evenly distributed. All evaluators had the task to annotate each assigned hazard at least once in each dataset (if present at all). The step to annotate all occurrences of individual hazards in all images is omitted as the required effort would exceed the resources reasonable for this proof of concept. The annotation tool is set to randomly choose the access order to reduce the impact of the individual image ordering and, thus, reduce the data base bias. Table 4 summarizes the results of the evaluation showing the number of images with hazards and the number of uniquely identified hazards. Not surprisingly, KITTI contains the most hazards. The reason is not only that it contains the most test cases, it is also created in the least controlled environment (outdoor street scenes). There are many deficiencies in recording quality manifesting as hazards and it includes images with motion blur as well as reflections on the windshield.

Many effects stemming from interactions of multiple light sources, medium effects, and sensor effects are missing in all three test datasets. The majority of identified hazards deal with specific situations that produce overexposure, underexposure, little texture, and occlusions.

6. Evaluation

In this Section we evaluate the effect of identified hazards on algorithm output quality itself. The example hazard evaluation from the last Section is used as a starting point. Evaluators annotated the test data to mark the areas in each image that correspond to a specific hazard. A coarse outline of relevant areas in an image is deemed to be sufficient for this evaluation, no pixel accurate labeling of the images is needed. A specific hazard can only impact the system

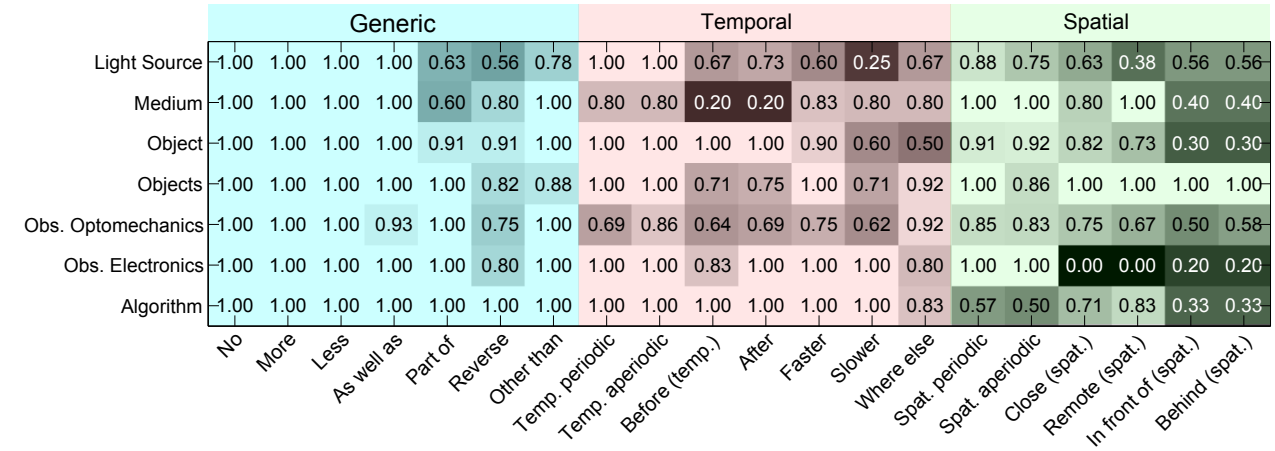

Figure 3. Ratio of meaningful combinations for each guide word per location (averaged over all parameters of each location)

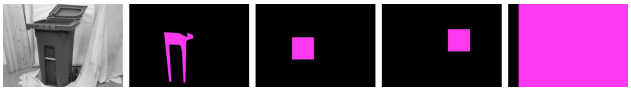

Figure 4. Example for annotation masks for hazard 'No Texture' (from left to right): input image, shape, box, rand, all

Dataset	SAD	CEN	SGBM	CVF	PM	SCAA
MB06	1.61	1.83	1.95	1.60	1.49	1.15
MB14	1.47	1.58	1.67	1.76	1.65	1.97
KITTI	1.17	1.68	1.84	1.29	2.23	1.05

Table 5. Ratio between the error threshold of *shape* and *all*

if it is visible in the image. We validate that the annotated area itself (and, thus, the included hazard) is responsible for the output quality decrease by adding four maskings as controls. The different masks represent a step-by-step increase of influence of the annotated areas:

- *shape*: masks with the annotated outlines as filled polygons,
- *box*: masks with boxes of equal size and center of gravity as each annotated outline,
- *rand*: masks with boxes of equal size as the annotated outlines but a randomly placed center of gravity,
- *all*: masks with all pixels except the left border region.

Figure 4 gives an example of the generated masks. The *rand* mask only represents the annotated area's size, *box* represents area and position while *shape* represents the full annotation. The *rand* vs. *all* masks verify if the output quality is affected by using smaller image parts for evaluation instead of the whole image, while *box* vs. *shape* evaluates the influence of specific shapes of the annotations.

Table 4 lists the resulting number of annotations created for each dataset. Some hazards require the selection of split areas, thus, resulting in multiple annotations. We calculate the amount of pixels covered by GT in our individual masks and only use these pixels for evaluation. Unfortunately, many of the hazards (*e.g.* reflections, transparencies, occlusions, very dark materials) also have a negative influence on the laser scanner used for the GT generation in KITTI. The GT data is generally sparse and even more sparse in the annotated areas.

For evaluation of the stereo vision test dataset we used the following popular stereo vision algorithms: SAD + Texture Thresholding (TX) & Connected Component Filtering (CCF) [21], SGBM + TX & CCF [13], Census-based BM + TX & CCF [15, 17], Cost-Volume Filtering (CVF) & Weighted Median Post Processing Filtering (WM) [30], PatchMatch (PM) & WM [4], and Cross-Scale Cost Aggregation using Census and Segment-Trees & WM [43, 25]. The resulting disparities of each stereo vision algorithm is compared to the GT disparities of the test dataset. The number of wrong pixels (error threshold of 2px) is then compared to the number of pixels within the respective mask that had valid ground truth values. Invalids in the result are counted as being above any threshold. Figure 5 shows the result of the evaluation for all three datasets.

6.1. Interpretation

The effect of applying the masks based on the identified hazards can be clearly seen. Table 5 summarizes the ratios between the error values of *shape* and *all*. The correctly masked areas (*shape*) have higher error ratios than the mean for the full image (*all*). The results for KITTI are much more erratic than the rest. The large amount of missing GT data in this dataset reduced its value for this evaluation drastically. The majority of *shape* mask areas have higher error ratios than the same-sized *box* mask areas. Newer and more complex algorithms generally score lower errors and have lower absolute differences between *shape* and *all* errors. There are two distinct groupings: *rand* masks have comparable results as *all* masks while *box* is comparable to *shape* [3]. This allows for the following conclusions

[3]This suggests that box annotations can often be used instead of the time-consuming shape annotations.

Algorithm	Image Pairs	Images w. Hazards	Found Hazards	# Annotations	% GT all	% GT rand	% GT box	% GT shape
Middlebury Stereo Evaluation (MB06) [32]	26	19	34	55	95.9	92.3	94.8	92.5
Middlebury Stereo Eval. "New" (MB14) [31]	23	17	57	80	96.9	94.8	93.4	91.2
The KITTI Vision Benchmark (KITTI) [10]	194	62	76	101	45.7	45.6	37.1	37.4

Table 4. Stereo vision test datasets used in our evaluation, number of found hazards and percentage of masks covered by GT

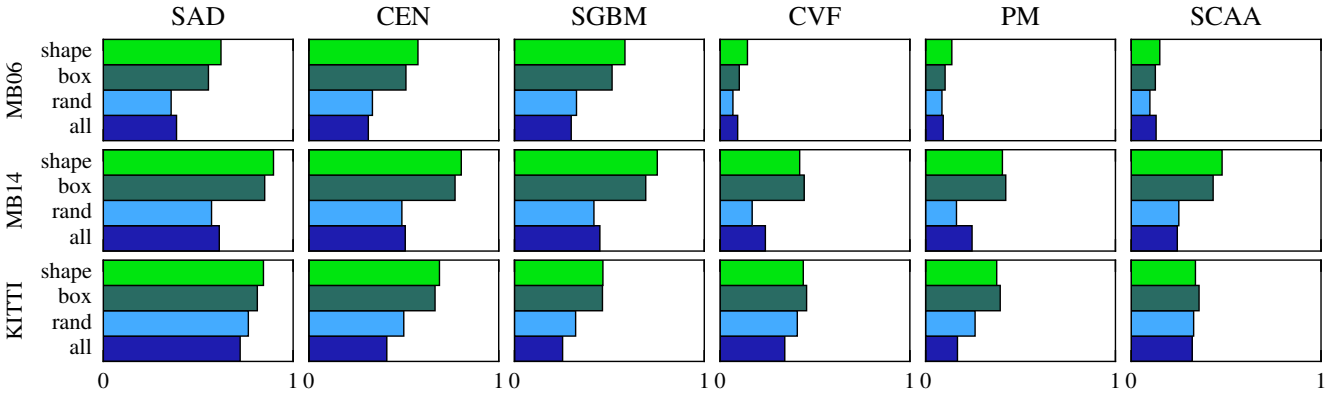

Figure 5. Percentage of pixels with an error above 2px for all algorithms (false positives) and the different applied maskings

based on the different maskings: algorithms have higher error rates at annotated areas and score even higher error rates if the annotation's shape is preserved (*shape* vs. *box*). The effect of sampling patches of different sizes in each image is not prevalent (*rand* vs. *box*) and can be neglected. The evaluation paints a clear picture: Areas identified by the CV experts as containing a visual hazard guided by the CV-HAZOP checklist are especially challenging to the CV algorithms. Thus, these challenging areas need to be focused on for robustness evaluations.

7. Conclusion

The creation of a checklist containing critical situations and relations that have the potential to reduce the quality and functionality of CV systems is a crucial component on the road towards systematic validation of CV algorithms. This paper presents the efforts of several experts from the fields of CV as well as risk and safety assessment to systematically create such a list. To the authors' best knowledge, this is the first time that the risk analysis method HAZOP has been applied extensively to the field of computer vision.

The CV-HAZOP is performed by first introducing a generic CV model which is based upon information flow and transformation. The model partitions the system into multiple subsystems which are called locations. A set of parameters for each location, that characterize the location's individual influence on information is defined. Additional special CV-relevant "guide words" are introduced that represent deviations of parameters with the potential to create hazards. The execution of the HAZOP is performed by a number of authors in parallel, assigning meanings to each combination of guide words and parameters to identify hazards. The individual findings are discussed and merged into one resulting CV-HAZOP list. A guideline for using the hazard list as a tool for evaluating and improving the quality and thoroughness of test datasets is provided.

The CV-HAZOP has produced a comprehensive checklist of hazards for the generic CV algorithm with over 900 unique entries. It supports structured analysis of existing datasets and calculation of their hazard coverage in respect to the checklist. We present an example by applying the proposed guidelines to popular stereo vision datasets and finally evaluate the impact of identified hazards on stereo vision performance. The results show a clear correlation: identified hazards reduce output quality.

Our HAZOP checklist is not considered final. It will be updated to include lessons learned during evaluations, testing or even after tested systems are put into operation. By sharing this information with the community over our public HAZOP database we hope for further improvements of quality and feasibility of the process and reduction of the effort needed for CV robustness evaluation. At this stage, the CV-HAZOP becomes a structured and accessible reference hub for sharing experiences with CV algorithm development, usage, and maintenance.

8. Acknowledgements

Special thanks for their extensive CV-HAZOP contributions go to Lawitzky G., Wichert G., Feiten W. (Siemens Munich), Köthe U. (HCI Heidelberg), Fischer J. (Fraunhofer IPA), and Zinner C. (AIT). Thanks to Cho J.-H. (TU Wien) and Beham M. (AIT) for their help with the example chapter. The creation of the CV-HAZOP as well as this work have been funded by the ARTEMIS project R3-COP, no. 100233.

References

[1] *Procedures for Performing a Failure Mode, Effects and Criticality Analysis, MIL-STD-1629A*. Department of Defense, 1949.

[2] *Reliability Prediction of Electronic Equipment: MIL-HDBK-217F*. Department of Defense, 1991.

[3] J. Y. Aloimonos and D. Shulman. *Integration of visual modules: an extension of the Marr paradigm*. Academic Press Professional, Inc., Boston, 1989.

[4] M. Bleyer, C. Rhemann, and C. Rother. Patchmatch stereo-stereo matching with slanted support windows. In *British Machine Vision Conference*, 2011.

[5] K. Bowyer, C. Kranenburg, and S. Dougherty. Edge detector evaluation using empirical ROC curves. *Computer Vision and Image Understanding*, 84(1):77–103, 2001.

[6] K. Bowyer and P. J. Phillips. *Empirical Evaluation Techniques in Computer Vision*. IEEE Computer Society Press, Los Alamitos, CA, USA, 1998.

[7] Center for Chemical Process Safety. *Guidelines for Hazard Evaluation Procedures, with Worked Examples, 2nd ed.* Wiley, 1992.

[8] P. Fenelon and B. Hebbron. Applying HAZOP to software engineering models. In *Risk Management And Critical Protective Systems: Proceedings of SARSS*, pages 11–116, 1994.

[9] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381395, 1981.

[10] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Computer Vision and Pattern Recognition*, 2012.

[11] K. Goseva-Popstojanova, A. Hassan, A. Guedem, W. Abdelmoez, D. E. M. Nassar, H. Ammar, and A. Mili. Architectural-level risk analysis using uml. *IEEE Trans. Softw. Eng.*, 29(10):946–960, Oct. 2003.

[12] F. R. Hampel. A general qualitative definition of robustness. *The Annals of Mathematical Statistics*, 42(6), 1971.

[13] H. Hirschmüller. Stereo processing by semiglobal matching and mutual information. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(2):328–341, 2008.

[14] P. Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35(1):73–101, 1964.

[15] M. Humenberger, C. Zinner, M. Weber, W. Kubinger, and M. Vincze. A fast stereo matching algorithm suitable for embedded real-time systems. *Computer Vision and Image Understanding*, 2010.

[16] *Functional safety of electrical/electronic/programmable electronic safety-related systems - Part 4: Definitions and abbreviations: IEC 61508-4*. International Electrotechnical Commission, 2010.

[17] T. Kadiofsky, J. Weichselbaum, and C. Zinner. Off-road terrain mapping based on dense hierarchical real-time stereo vision. In *Advances in Visual Computing*, volume 7431 of *Lecture Notes in Computer Science*, pages 404–415. Springer Berlin Heidelberg, 2012.

[18] J. T. Kajiya. The rendering equation. *SIGGRAPH Conference Proc.*, 20(4):143–150, Aug. 1986.

[19] T. A. Kletz. HAZOP and HAZAN notes on the identification and assessment of hazards. *The Institution of Chemical Engineers*, 1983.

[20] D. Kondermann, R. Nair, S. Meister, W. Mischler, B. Güssefeld, K. Honauer, S. Hofmann, C. Brenner, and B. Jähne. Stereo ground truth with error bars. In *Asian Conference on Computer Vision*, 2015.

[21] K. Konolige. Small vision systems: Hardware and implementation. In *Robotics Research*. Springer, 1998.

[22] J. Laprie. *Dependability: Basic Concepts and Terminology.*, volume 5 of *Dependable computing and fault-tolerant systems*. Springer, 1992.

[23] D. Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. W. H. Freeman, 1982.

[24] B. Matthias, S. Oberer-Treitz, H. Staab, E. Schuller, and S. Peldschus. Injury risk quantification for industrial robots in collaborative operation with humans. In *Proc. of 41st International Symposium on Robotics and 6th German Conference on Robotics*, 2010.

[25] X. Mei, X. Sun, W. Dong, H. Wang, and X. Zhang. Segment-tree based cost aggregation for stereo matching. In *Computer Vision and Pattern Recognition*, pages 313–320, 2013.

[26] J. Min, M. Powell, and K. W. Bowyer. Automated performance evaluation of range image segmentation algorithms. *IEEE Transactions on Systems Man and Cybernetics - Part B - Cybernetics*, 34:263–271, 2004.

[27] H. Nyquist. Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers*, 47(2):617–644, 1928.

[28] N. Pinto, D. D. Cox, and J. J. DiCarlo. Why is real-world visual object recognition hard? *PLoS Comput Biol*, 4(1), 2008.

[29] J. Ponce, T. L. Berg, M. Everingham, D. A. Forsyth, M. Hebert, S. Lazebnik, M. Marszalek, C. Schmid, B. C. Russell, A. Torralba, et al. Dataset issues in object recognition. In *Toward category-level object recognition*, pages 29–48. Springer, 2006.

[30] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. In *Computer Vision and Pattern Recognition*, pages 3017–3024, 2011.

[31] D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nesic, X. Wang, and P. Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In *Pattern Recognition*, pages 31–42. Springer, 2014.

[32] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International journal of computer vision*, 47(1):7–42, 2002.

[33] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. In *Computer Vision and Pattern Recognition*, 2003.

[34] R. Schlick, W. Herzner, and E. Jöbstl. Fault-based generation of test cases from uml-models approach and some experiences. In *Computer Safety, Reliability, and Security*, volume 6894 of *Lecture Notes in Computer Science*, pages 270–283. Springer Berlin Heidelberg, 2011.

[35] C. E. Shannon. Communication in the presence of noise. *Proc. of the Institute of Radio Engineers*, 37(1):10–21, 1949.

[36] C. E. Shannon and W. Weaver. *The Mathematical Theory of Communication*. University of Illinois Press, USA, 1949.

[37] C. Strecha, W. von Hansen, L. Van Gool, P. Fua, and U. Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *Computer Vision and Pattern Recognition*, 2008.

[38] A. Takanen, J. DeMott, and C. Miller. *Fuzzing for software security testing and quality assurance*. Artech House on Demand, 2008.

[39] N. Thacker, A. Clark, J. Barron, J. Ross Beveridge, P. Courtney, W. Crum, V. Ramesh, and C. Clark. Performance characterization in computer vision: A guide to best practices. *Computer vision and image understanding*, 109(3):305–334, 2008.

[40] A. Torralba and A. A. Efros. Unbiased look at dataset bias. In *Computer Vision and Pattern Recognition*, pages 1521–1528, 2011.

[41] R. Van der Spek and A. Spijkervet. Knowledge management: Dealing intelligently with knowledge. *Knowledge management and its integrative elements*, pages 31–60, 1997.

[42] W. E. Vesely, F. F. Goldberg, N. H. Roberts, and D. F. Haasl. *Fault Tree Handbook*. Systems and Reliability Research, Office of Nuclear Regulatory Research, U.S. NRC, 1981.

[43] K. Zhang, Y. Fang, D. Min, L. Sun, S. Yang, S. Yan, and Q. Tian. Cross-scale cost aggregation for stereo matching. In *Computer Vision and Pattern Recognition*, 2014.