Risk analysis for smart homes and domestic robots using robust shape and physics descriptors, and complex boosting techniques

Rob Dupre[a], Vasileios Argyriou[a], George Tzimiropoulos[b], Darrel Greenhill[a]

[a]*Faculty of Science, Engineering and Computing, Penrhyn Road, Kingston upon Thames, Surrey KT1 2EE, UK*
[b]*School of of Computer Science, University of Nottingham, Nottingham, NG8 1BB, UK*

Abstract

In this paper, the notion of risk analysis within 3D scenes using vision based techniques is introduced. In particular the problem of risk estimation of indoor environments at the scene and object level is considered, with applications in domestic robots and smart homes. To this end, the proposed Risk Estimation Framework is described, which provides a quantified risk score for a given scene. This methodology is extended with the introduction of a novel robust kernel for 3D shape descriptors such as 3D HOG and SIFT3D, which aims to reduce the effects of outliers in the proposed risk recognition methodology. The Physics Behaviour Feature (PBF) is presented, which uses an object's angular velocity obtained using Newtonian physics simulation as a descriptor. Furthermore, an extension of boosting techniques for learning is suggested in the form of the novel Complex and Hyper-Complex Adaboost, which greatly increase the computation efficiency of the original technique. In order to evaluate the proposed robust descriptors an enriched version of the 3D Risk Scenes (3DRS) dataset with extra objects, scenes and meta-data was utilised. A comparative study was conducted demonstrating that the suggested approach outperforms current state-of-the-art descriptors.

Keywords: 3D Scene analysis, Risk Estimation, Domestic robots, Smart homes, HOG, 3D VHOG

2010 MSC: 00-01, 99-00

1. Introduction

Scene analysis is a research area spanning a large range of topics, both indoor and outdoor, with applications in navigation systems [42], traffic analysis [6, 7], domestic robotics [46], smart homes [9] and more recently the concept of risk detection [18, 57] amongst many others. In this work the problem of evaluating risk for indoor applications is considered, more specifically mimicking a human's ability to analyse and identify risks. To this end a quantified risk score for 3D scenes using vision based techniques is provided. The concept of risk assessment is derived from the ability of humans to identify a potentially hazardous environment using a range of attributes, evaluating those specific characteristics based on experience and determining whether a threat is present or not [5].

The definition of what can be considered a risk or hazard in an environment is contextual. What can be considered safe in one environment may not be in others. For example a container of liquid at the edge of a table is risky in a household environment, however in a lab this might pose a far larger danger. Similarly users of the environment will also effect how risk is perceived, if the environment contains children or elderly adults the threshold of what is risky may need to change. However regardless of context, the elements that might contribute to the concept of risk can be broken down into components from which a decision can be made. These components include elements such as shape, size, material, temperature, position and many others. With this risk analysis functionality domestic robots could be trained to help avoid potentially hazardous situations. In the Smart Home example; attention could be drawn to these situations and accidents avoided.

The Risk Estimation Framework [17] measures risk as a function of measurable elements in a scene, the methodology relies on a combination of 3D shape descriptors and Newtonian physics based on supervised learning. Firstly, at a global level, the scene is analysed holistically using the concept of scene stability. For example, classifying a glass bottle in the corner of a table as more

(a) (b) (c)

Figure 1: (a) Example scene with objects demonstrating a variety of intrinsic properties (e.g. sharp, pointed), (b) scene with a variety of stability levels, and (c) a scene reconstructed using Kinect Fusion before and after the plane removal.

hazardous than the one placed at the centre (Figure 1b). Secondly, the scene is analysed at a local level, looking to identify "hazard-related" shape features of objects within the scene. Here the term feature relates to an actual physical property of an object (e.g. sharp, pointed). As an example a knife would have a sharp blade, which would be classified as a "hazard" feature (Figure 1a). We emphasize that in this system the problem of object recognition is bypassed and only local object properties are recognised, allowing the proposed approach to be more flexible and generic. Additionally this overcomes the problem of similar object classes containing objects which might have different levels of risk, for example a steak knife compared to a butter knife. As with all local level features a model of "hazard features" from a training set is constructed and used to test future unknown examples.

This work is an extension of the paper [17] and introduces the following contributions. A) the novel robust kernel for 3D descriptors in comparison to the work in [18], B) an advanced boosting mechanism that supports complex

data for supervised learning, C) a novel shape descriptor based on Newtonian Physics and D) an enriched version on the 3DRS data set. In more details; the robust kernel for 3D descriptors is suggested, which can reduce the effects that outliers have in the supervised learning mechanisms. Secondly, Complex and Hyper-Complex variants of Adaboost [21] are presented, which provide an increase in computational efficiency. Thirdly, the Physics Behaviour Feature (PBF) descriptor is introduced utilising the physical properties of an object to identify hazardous objects. This is achieved through the application of Newtonian Physics and the estimation of an object's angular velocity after the application of a force. Our final contribution is the enriched version of the 3D Risk Scenes (3DRS) dataset with additional objects, meta-data and risk scenes to create a more challenging and complete dataset for 3D scene risk analysis.

The paper will continue as follows; in section 2 an analysis of the similar areas of research will be followed by an overview of related work. The proposed methodologies and contributions used in this work will be presented in section 3. Section 4 will outline our comparative study with other state-of-the-art methods and analyse the results. Finally, in section 5 conclusions are drawn.

2. Related work

The following section provides an overview of existing work in scene analysis with respect to risk assessment, followed by a review of existing feature descriptors relevant to the methodology.

2.1. Scene analysis and risk assessment

Risk assessment for a given environment finds applications in many areas, from workplace health and safety to analysis of disaster zones to name a few. With the advent of consumer available depth acquisition hardware [25] and laser scanning systems such as LiDAR, research in scene analysis in the 3D domain has grown considerably [8, 35, 54].

In [57, 58], the authors analyse a scene based on the probability of an object being dislodged using disturbance fields. By modeling human actions and natural events such as earthquakes or wind effects, the probability of objects falling can be calculated. This yields a risk score based on a specific type of input, which requires modeling per event. Additionally, their approach does not take into account the possibility that objects may collide with each other, nor is any weighting given to the risk of the object itself.

Other existing work on risk assessment exists in similar areas such as patient monitoring [45, 3], where the focus is on indoor fall assessment for elderly adults. Though conceptually similar, these papers focus on analysing the risk associated with the persons and not their environment. Work on robotics for medical applications [12] defines safety zones around anatomical areas, such as major nerval and vascular structures. This prevents the robotic system entering these zones, providing an efficient way of preventing injury. However, the system does not apply reasoning to the environment. Additionally although the system tracks patients movement, it requires pre-programming for each change in situation.

With advances in the industrial robotic sector and robotic hardware, new areas of risk in various workplaces have been identified. In [14, 34], a review is provided into these hazards and the principles of guarding to ensure human safety. Hazard analysis, safety precautions, programming procedures and maintenance of the robots are also discussed.

Finally, with advances in robotics and unmanned drones, the functionality to fully automate these devices using vision based techniques is emerging [42, 55]. Though these proposed systems do not emphatically determine risk, they do analyse the environment to identify a suitable landing zone based on a set of parameters.

Another emerging area of research within scene analysis relates to 3D volumetric reasoning. Which provides a better understanding of a scene by analysing properties such as its overall stability, for example whether objects within the volume are supporting one another. This draws heavily from the human ability

to analyse a scene and make fast judgements about the environment. Battaglia et al. [2] explores this concept and introduces the idea of an "Intuitive Physics Engine (IPE)" that attempts to mimic human cognitive simulation process when analysing a scene. Wu et al. [56] extends this principle by incorporating a physics engine with representation learning. Their work further supports the idea that a humans ability to analyse a scene is based upon a realistic physics engine as part of a generative model to interpret real-world physical scenes. Additionally the system is also capable of outputting physical properties of objects from video observations such as mass and friction coefficients. Although the concept of risk in the environment is raised in some of this work, an automated form of risk evaluation for a given scene is not addressed.

2.2. 3D local descriptors

Within the proposed work, three dimensional descriptors are proposed and as such an overview of existing research is given. Arguably, the advent of SIFT [32] and HOG [11] revolutionized 2D object recognition by creating a local descriptor that was robust to geometric and photometric changes. With the advent of cheap 3D depth camera hardware, such as the Microsoft Kinect [43], work has been done to transfer HOG [17], SIFT [41, 23], Harris [44] and FAST [38] into 3D.

Scherer et al. [40] does gradient computation in 3D using a convoluted distance field. This provides an effective way of calculating the magnitudes of the gradients, scoring them highly when localised near a surface of a model (local maxima), however their method also scores highly those at local minima creating additional artifacts within the data. As such this particular implementation is unsuitable for our local feature recognition.

Tang et al. [47] presents the Histogram of Orientated Vectors (HOVN) feature. Here the normal vectors are used as the features to capture local geometric characteristics which is used for object recognition. Another method, which extends HOG to 3D, is presented in [28, 36]. In particular, HOG is extended through the use of time as the third dimension. This allows the creation of spa-

tiotemporal features that can be used for action recognition in video sequences. This approach is based on 2D image based intensity gradients without taking into account concepts related to the density of an area and therefore it is not an appropriate descriptor for objects with non-uniform density.

Tombari et al. [49] examine local 3D descriptors and define two main categories in which they fall; signatures and histograms. Signatures are potentially highly descriptive through the use of spatially localized information. Whereas histograms sacrifice descriptive power for robustness through compression of geometric structure into bins. The Signature of Histograms of Orientations (SHOT) feature is presented, which encodes histograms of the normals of the points within a neighbourhood as well as introduces geometric information concerning the location of the points within that neighbourhood.

Frome et al. [22] utilise 3D shape and Harmonic shape contexts to build a feature descriptor to find cars in point cloud data. The feature descriptors are defined for an arbitrary set of basis points within the point cloud and are compared using distance measures, such as L2, to a predefined reference set. The methodology is demonstrated on an extensive car database in both the presence of clutter and noise.

Cirujeda et al. [10] presents a descriptor based on the covariance of features, combining shape and color information of 3D surfaces. Multi-scale covariance descriptor (MCOV) has a number of properties including; invariant to spatial rigid transformations, robust to noise and resolution changes and is applicable to characteristic point detection. Additionally, features are defined using a multi-scale framework, which helps link the various features not only on a local scale but additionally at a more global level too. This has the advantage of reducing repeatability problems and improving detection of points in edges or borders of scene objects.

Rusu et al. [39] proposes an extension to their already well known Point Feature Histograms (PFH) in the form of Fast Point Feature Histograms (FPFH). FPFH is considerably faster and can be computed online due to a reduction in computational complexity to $O(k)$ (over $O(k^2)$ for PFH) whilst retaining most

of the descriptive power of the PFH.

Flint et al. [20] combines the advantages of SIFT descriptor and the SURF detector to produce the ThriIFT 3D feature detector. ThrIFT utilises 3D Hessians and creates a weighted histogram of the deviation angles between the normals of points in the neighbourhood of the original feature point.

Finally, the work in [16] uses point pair features to define global model descriptors aiming to recognise similar objects within a point cloud scene. The feature is based on the distance between the point pair, the angles from surface normal to point pair line, and finally the angle between the two normals. Then using a voting system, it matches pre-defined features to objects in a scene. This system presented good results for object recognition, but operates on a global scale, making it unsuitable for the concept of specific local feature recognition. Additionally the work in [33] is also worth mentioning at this point.

3. Proposed methodology

The following section discusses the Risk Estimation Framework, and in detail the proposed robust kernel for 3D descriptors and the complex and hypercomplex Adaboost methodologies. In Figure 2 the proposed methodology is illustrated, outlining the end to end solution and where each of the proposed techniques fit. Initially the given scene is preprocessed to provide individual object clusters. Using these object clusters the stability of each object is estimated, providing one element of the risk score. The hazard features of each object cluster are then analysed, using the 3D Voxel HOG and Physics Behaviour Feature, the results of which are used as the second element of the risk score. More detail for each aspect of the frame work is given below.

3.1. Pre-processing

Before the risk in a scene can be evaluated some pre-processing steps are required to change the data into a suitable format. Figure 3 demonstrates this process.

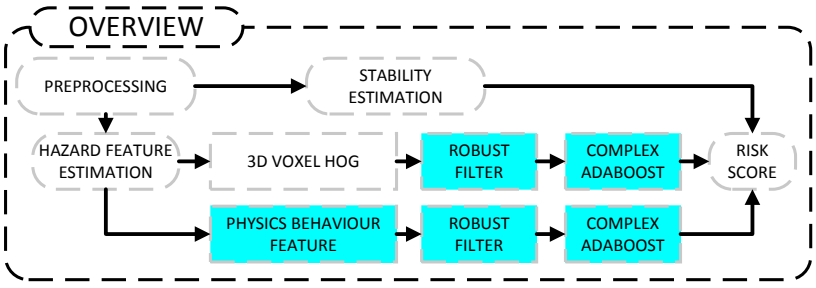


Figure 2: The overall methodology for the Risk Estimation Framework, with each of the newly proposed methodologies highlighted.

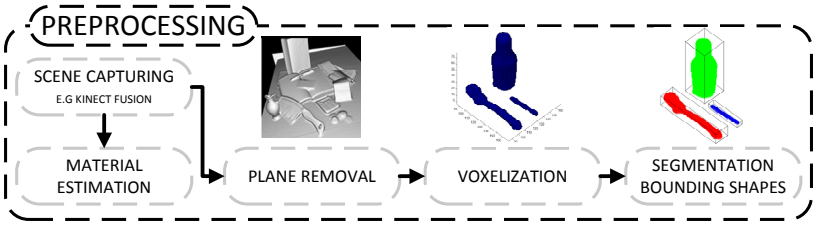


Figure 3: Pre-processing steps: Scene capturing with Kinect Fusion, Plane removal, voxelization and segmentation.

Scene data and 3D mesh model reconstruction are assumed to be captured using methods such as Simultaneous Mapping and Localisation (SLAM) techniques e.g Kinect Fusion [25] or multi camera acquisition systems [52]. Additionally other sensors such as thermal or acoustic cameras could also be used. Each method returns a three dimensional representation of the subject scene, either in an already voxelized form, point cloud format or as a vertex/face based 3D model. In this work scenes have been captured using Kinect Fusion, using a Kinect camera. This returns a point cloud representation of the scene.

The surface on which the objects are set requires removal prior to segmentation. In the case of the given scene this represents the table surface on which the objects in a scene are set. The work by [50] presents a solution to this using connected component based clustering in point clouds together with a 'planar refinement step'. The dimensions of the removed plane are recorded and used later to define the surface during simulation.

The returned 3D model is then requires conversion to a data format that is suitable for use in the provided methodology. Voxelization is used to produce an equally spaced grid representation of the scene, where each voxel provides a binary classification of either object or not. For this process we rely on existing techniques based on the work in [24]. Initially a grid is defined in 3D space around the model. Using the vertices of the model with a defined radius, voxels who's centre falls into this area are defined as part of the model. Using the edge information a cylinder is defined along the length of the edge, voxels who's centre falls into the area of the cylinder are also classified as part of the model. Finally for a given face of the model, two additional planes are defined above and below the surface of the given face and all voxels who's centres lie within this area are attributed to the voxel representation of the model. At each stage of this process a rule is applied to the voxel that helps maintain a hole free voxel surface. The rules define relationships to neighbouring voxels based on the model data that is used to define it. Additionally a voxel representation is also optimised based on principles of accuracy, minimality and separability. Where accuracy is a defined measure to quantify how well represented the model is, separability which could be described as the appropriate separation of voxel space using the defined voxel surface and finally, minimality, which ensures that additional voxels are removed subject to accuracy and separability. Voxels which are enclosed within a mesh, are also classified as part of the object allowing the consideration of features based on an object's density. This step may be avoided if the data capture method returns a voxelized representation of the scene [27].

With this representation of the scene, clustering of the voxel volume can be applied. A number of different clustering algorithms were tested, using modified versions of the work presented in [50, 15]. A bounding box for each object cluster is defined, the dimensions of which are based on the returned clusters.

To represent the scene objects within a physics simulation, utilised in sections 3.3 and 3.4.1, a range of bounding shape primitives (e.g. box, cylinder, sphere, etc.) can be used. The shape primitive that when fully encasing the cluster has the least empty voxels is the one that best defines the object cluster.

Additionally these bounding shapes must not intersect; as such a recursive reduction process is applied resizing bounding boxes until no overlap is detected. The result is a pre-processed scene in which each detected object cluster is assigned its own bounding shape.

3.2. Risk Estimation Framework

A cumulative risk score R for a scene is defined as the weighted sum of n measured risk elements E (1). The weighting specified for each element should fall into a range of zero to one, with the sum of all weightings being equal to one. A risk element is any measure that could highlight potential risk. These elements could include concepts such as stability, hazard shape features or any other properties that may present a danger, for example temperature obtained from a thermal camera or material analysis data. Each of these elements has an assigned weight; this allows the context of the risk to be considered, applying more weighting to elements that are more relevant in a given situation. For example, in an environment with adults present, stability may not have a weighting as high as in situations where children are present.

$$R = \sum_{i=1}^{n} (w_i E_i) \tag{1}$$

For the purpose of this paper we define the cumulative risk score R as a function of the weighted elements of stability S and hazard shape features H.

$$R = w_S S + w_H H \tag{2}$$

3.3. Stability Estimation

The proposed methodology for scene stability estimation is based on the use of Newtonian physics mechanics applied to the preprocessed scenes. To evaluate the stability of an object we replicate the application of forces from a variety of directions. Consequently, statistical analysis on the subjects of a simulation

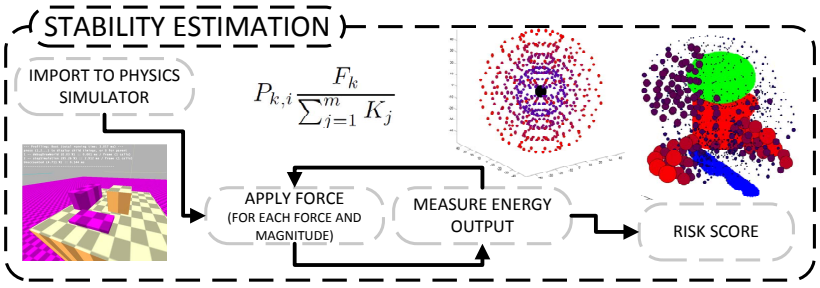


Figure 4: Stability estimation flow. Scene objects are imported into the physics simulation. Forces are applied from a sample of directions to each object in the scene, subject to (4). The energy output from each applied force is recorded. Simulations are repeated with forces of increased magnitude. For each object the resultant energy from each simulation is used to build a stability plot. The sum of all resultant energy defines the stability of the object and by extension its risk score.

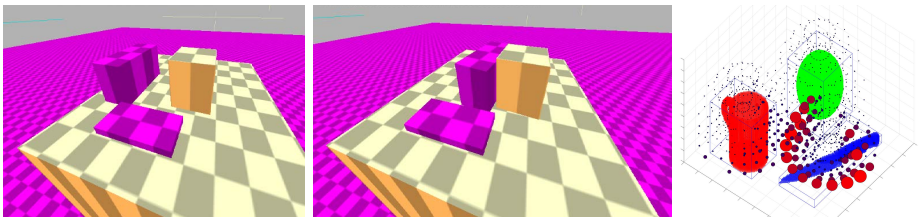


Figure 5: Stability evaluation process using Newtonian physics (Left) Initial layout in the physics simulation; (Middle) Collision occurring during the simulation; and (Right) Stability plot with the circles around the objects indicating the direction of instability with radius corresponding to the severity.

can be performed allowing us to compute the energy output from each applied force. An overview of this is presented in Figure 4.

Using 'collision shapes', in this case bounding boxes, the objects are recreated using simplistic primitives, which represent the overall shape. This reduces the computational costs needed to emulate its behaviour whilst maintaining a reasonable level of accuracy. To simulate an object's behaviour; parameters such as position, size, mass, friction and angular dampening coefficients are attached to these shapes. The bounding shape calculated during preprocessing serves as the guidelines for the collision shape, (position and size).

The surface the objects are placed on within the simulation is defined using dimensions obtained during the plane removal process in preprocessing. Mass is defined by calculating the number of voxels within an object cluster and using the assumption that all objects are made from the same material. However through the use of material estimation (such as BRDF function estimation [53, 29] or techniques such as visual vibrometry [13] as well as others [8, 56]), more accurate values for mass could be acquired for use in the simulations. Additionally with a defined material, the friction coefficients can be better estimated and applied to the simulation. These techniques would be applied during pre-processing (figure 3), however this falls into a separate area of research and is not the goal of this work, therefore global values are used for these parameters.

Stability s for a force k on a given object i is defined as the ratio of the applied force F_k over the summed kinetic energy K_j for all objects m in the scene. This is scaled by the possibility $P_{k,i}$ of the force being applied.

$$s_{k,i} = P_{k,i} \left(\frac{F_k}{\sum_{j=1}^{m} K_j} \Delta x \right) \tag{3}$$

where $K_j = \sum_{t=1}^{T} \frac{1}{2} M {V_t}^2$ represents the accumulated kinetic energy produced by the object j over time T as a result of the force k being applied during the simulation, obtained using numerical integration. Here M represents mass and V the velocity of the object j at a given time t. Δx is an object's displacement, but since the kinetic energy is calculated numerically over fixed length intervals, this value is equal to one.

Possibility $P_{k,i}$ represents the likelihood of a given force F_k being applied to object i. This is defined as whether the force could collide with the object without hitting first another entity within the scene. For example forces from below an object on a plane would collide with the surface first, therefore would not be considered.

$$P_{k,i} = \begin{cases} 1, & \text{if } F_k \textit{ directly collides with object } i \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

Forces of different strengths are applied to the center of each collision shape (object) during the simulation. The strength of these forces is widely sampled to ensure that objects of both large mass and small are effected and provide a measurable energy output. The force direction (angles) is selected uniformly over a sphere.

The resultant overall kinetic energy K for each object j is calculated. By analysing the amount of kinetic energy produced by each object for each force F, we can ascertain if, during the course of that simulation, an object has been dislodged from the surface or if other objects within a scene have been affected due to collision. By varying the strength of force we build up a picture of how unstable an object is in its environment. The total stability S of a scene is given as the sum of the estimated stability s for each force k applied to each object j.

$$S = \sum_{k=1}^{r} \sum_{j=1}^{m} s_{k,j} \tag{5}$$

The outcome of this allows the differentiation between the case of an object (e.g. glass bottle) being placed at the center of a table or at the edge, evaluating with enough precision the stability of each scene (Figure 5).

3.4. Hazard shape descriptors

The following sections outline in detail the proposed descriptors used within the Risk Estimation Framework to evaluate the hazardous properties of an object within a scene.

3.4.1. Physics Behaviour Feature (PBF) as a Shape Descriptor

Using the behaviour of an object within a simulation environment as a feature descriptor is a novel concept. Based on the values generated from a physics

simulation a feature vector can be constructed and a classification made relevant to its risk. The essence of the methodology is to define a feature descriptor that describes how each individual object acts when a force is applied. In Figure 6, an overview of how this feature is incorporated into the Risk Estimation Framework is presented.

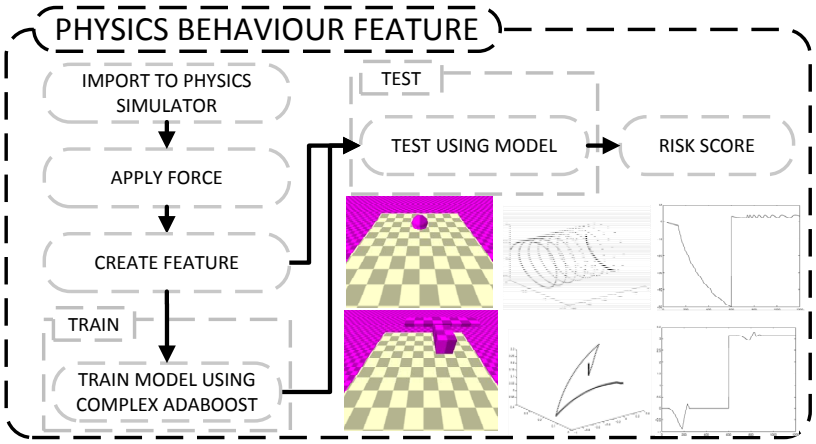
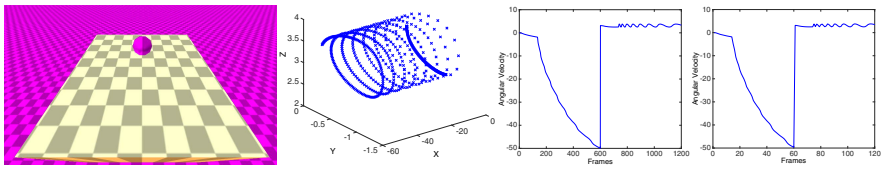


Figure 6: Physics Behaviour Feature flow. Initially an object is imported into the simulation environment. A single force is applied to the object and the position and rotation information is recorded. A feature vector is constructed and a model trained using Adaboost. The process is repeated with a new unknown object and, using the previously defined model, a classification as either hazardous or safe is returned.

Once pre-processing has been performed, an individual bounding shape for an object is passed to the physics engine. The goal is to take a single force from a fixed direction with a fixed magnitude and apply it to each individual object. The proposed feature descriptor is made up of the resultant simulation output data with reduced dimensionality.

For a given object x, force is applied to its bounding shape and it's angular velocity ω (in terms of x, y, z) over the duration of the simulation t is recorded. A feature vector is constructed from this data utilising dimensionality reduction to reduce three dimensions to two, additionally the data is sampled at a rate of one in ten to reduce the length of the final vector. The resultant feature vector corresponds to the physical and shape characteristics and properties of



(a) (b) (c) (d)

Figure 7: Physics Behaviour Feature, overview of the feature extraction process. (a) Simulation run on object bounding shape, angular velocity captured per frame, (b) The 3D plot of collected data, (c) Data reduced into 2D space and (d) down-sampled to the final feature vector (ω) without any significant loss of information.

an object in a scene.

$$\vec{x}^{\omega} = \{\boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_t\} \tag{6}$$

These features are used to create a decision model from supervised learning. A binary classification is returned defining the object as either being hazardous (1) or not (0). A confidence score based on the model's assessment can be used as a weighting to the binary classification. These values contribute to the hazard shape risk element as specified in (2).

3.4.2. A novel robust kernel for 3D shape descriptors

Other descriptors that could be used to identify hazardous objects based on their intrinsic properties (e.g. sharp, pointed) are 3D local shape features such as 3DSIFT, 3DHOG, 3D Voxel HOG [18], etc. Supervised learning techniques are utilised to classify the objects as risky or not, but due to noise of the RGBD acquisition devices and their low resolution the obtained accuracy is effected significantly. As a result of this, careful attention must be given to the outliers ensuring that the classification accuracy is reliable and remains as high as possible. In the following analysis the robust kernel for 3D local descriptors is outlined using 3D Voxel HOG [18] as an example, however the process is applicable to any descriptor without any modifications.

Traditional HOG applies a gradient vector to each pixel in an image in either one or both of the horizontal and vertical directions. The image is then divided into overlaying blocks, which in turn are made of a number of cells that contain a set number of pixels. For each cell, a histogram is created with evenly spaced bins representing gradient angles. Each pixel's gradient angle votes for a bin, with the contributed value being weighted in some way, usually utilising the gradient magnitude. Finally each block of cell histograms is normalised locally to reduce the impact of changes in illumination and a concatenation of the histogram values is used as the final feature vector.

This process is extended to the third dimension though the use of voxels. The process begins by breaking the voxel volume up into set feature blocks f comprised of a number of cubic 3D cells c, which in turn are made up of voxels v. Both number of blocks in a feature and number of voxels in cell is found experimentally and depends largely on the resolution of the 3D scans. For each voxel v within a cell the filter mask [-1,0,1] is applied on its neighbouring voxels in all three dimensions giving us the gradient vector $\vec{g}$ and its magnitude $\|\vec{g}\|$. Finally a weighting w is computed based on the gradient magnitude $\|\vec{g}\|$ and the total number of voxels in the cell c. The resultant 3D HOG histograms can present a way of identifying different types of features and intrinsic properties within an object. The same concept is applied to all the other 3D descriptors following their original implementations but allowing them to handle voxelised objects.

Let $\vec{x}^{3D}$ be the p-dimensional vector obtained by applying the 3D Voxel HOG (3D VHOG) in an area of a given scene. Based on the work in [19] on robust correlation translation estimation, the L_2-norm is replaced with the dissimilarity measure below:

$$d(\vec{x}^{3DVHOG}, \vec{x}_q^{3DVHOG}) = \sum_c \{1 - cos(a\pi(\vec{x}^{3DVHOG} - \vec{x}_q^{3DVHOG}))\} \quad (7)$$

where the values of the corresponding 3D VHOG features $\vec{x}^{3DVHOG}$, $\vec{x}_q^{3DVHOG}$ are represented in the range $[0, 1]$. A small value for α results in a function which resembles the L_2-norm. With increasing α, the effect of large distances possibly caused by outliers is reduced. In general, α represents the frequency of the cosine and is optimized to suppress the values caused by outliers. This kernel can be represented using the Euler form of complex numbers. In more detail, the angle values of $\vec{x}^{3DVHOG}$ normalised in $[0, 1]$ are mapped onto the complex representation $\vec{z}^{3DVHOG}$

$$\vec{z}^{3DVHOG} = \frac{1}{\sqrt{2}} e^{ia\pi\vec{x}^{3DVHOG}} \tag{8}$$

The values of $\vec{z}^{3DVHOG}$ will be now considered the feature vector used in our learning mechanism. The proposed robust 3D VHOG is a descriptor feature refinement, which aims to reduce the effects of these outliers. The same kernel can be used without any modification by the other descriptors such as 3D SIFT, 3D HOG and 3D Harris.

The pseudo code for the robust 3D VHOG implementation is outlined below.

```
1. choose Size of Cell and Feature Block
2. FOREACH Voxel v DO
3.    compute Weight w, GradientVector(g),
        Vector Magnitude ||g||
   end
4. FOREACH Cell c in Feature Block f DO
5.    create blockHistogram(θ_bins, φ_bins)
6.    FOREACH voxel v in c DO
7.       insert w||g|| into blockHistogram(θ, φ)
      end
   end
8. L2Normalize(blockHistogram in Feature)
9. RobustKernel(Feature)
end
```

3.5. Complex and Hyper-Complex Adaboost

This part of the proposed framework for risk estimation and scene analysis concerns the classification process which is based on supervised boosting techniques. In this section a novel extension of Adaboost is proposed to handle complex or hyper-complex feature vectors such as those produced by the proposed robust kernel for the 3D VHOG descriptor or any other similar one.

3.5.1. Learning via Boosting

The motivation for the proposed complex Adaboost comes from the proposed robust descriptor. The descriptor encodes histograms as angular data of the form $z = \cos(a) + j\sin(a)$. In this space, to measure similarity a Hermitian inner product between two descriptors z_1 and z_2 can be defined as $z_1^H z_2$. Although one can replace this with a concatenation of the cosines and sines of the form $x = [\cos(a); \sin(a)]$ and then measure similarity using the familiar inner product $x_1^T x_2$, this implies assuming independence between the elements of the feature vector. This assumption is not always valid, and although commonly accepted, it may lead to a loss of discriminative richness of the vectorial features [1, 31], which can be exploited further by considering the correlation information between the components.

Adaboost is a learning technique that creates a non-linear classifier to separate data into two groups. Weak classifiers are defined with a final strong classifier being a combination of these. At each iteration the weak classifiers with the lowest error margin are used to define the next in a 'greedy fashion'. Regarding the proposed features in both cases given N training examples $(\vec{x}_1, ..., \vec{x}_N)$, the corresponding labels $(y_1, ..., y_N)$ with $y_i \in \{-1, 1\}$, and an initial distribution of weights $W_1(i)$ a strong classification model $H(x)$ is obtained based on the weak classifiers h.

The weak classifiers are trained over a number of iterations Q using the weights' distribution W_t aiming to minimize ϵ_t, defined as the weighted sum error for misclassified points $\epsilon_t = \sum_i w_{i,t} e^{-y_i h_i \alpha_t}$ with α to be the minimizer of the exponential error function. In each iteration the error ϵ_t is estimated based

on the current weights W_t, which are updated before the next iteration.

$$W_{t+1}(i) = \frac{W_t(i)\exp(-a_t y_i h_t(x_i))}{Z_t} \tag{9}$$

where $a_t = -\frac{1}{2}\log(\frac{\epsilon_t}{(1-\epsilon_t)})$ and $Z_t = 2\sqrt{\epsilon_t(1-\epsilon_t)}$ is a normalization factor. The strong classifier is defined as $H(x) = sign(f(x))$, where $f(x) = \frac{\vec{a}\cdot\vec{h}(x)}{\|\vec{a}\|_1}$.

Regarding the boosting approach, because of the way weak classifiers are selected a complicated feature problem can be broken down and classified using a sparse classification rule, based on only a few features. This makes computation much faster as only a subset of the features are used. This is essential if the methodology is to be implemented in a real time scenario.

Finally, in order to define the second element H of the risk score R in (2) related to the 'hazard intrinsic features' the obtained outcomes from the classification process above are utilised.

$$H^{3D} = \frac{1}{m}\sum_{j=1}^{m}\left(\frac{\sum_{k=1}^{M} w_D G(j,k)}{\sum_{k=1}^{M} G(j,k)}\right)$$
$$H^{\omega} = \frac{1}{m}\sum_{j=1}^{m} w_D(j) \tag{10}$$

where $w_D = f(x)$ normalised and $G = \frac{1}{2}(sign(f(x)) + 1)$. As it is shown in (10), the confidence score obtained from Adaboost is used to evaluate the risk level of the scene and the objects.

As in our setting both the objects as well as their locations are known, we opted for a discriminative approach based on robust descriptors extracted from the objects of interest and supervised learning using complex Adaboost instead of a bagging approach.

3.5.2. Complex and Hyper-Complex Adaboost

In this section we present Complex and Hyper-Complex Adaboost, which implement a modification to the traditional Adaboost utilising complex numbers for use within weak classifiers suitable for the proposed robust kernel. In

Adaboost, each weak classifier h_t must determine the optimum threshold per feature dimension that minimises the classification error ε_j, as described in (11).

$$h_t = \arg\min_{h_j \in H} \varepsilon_j = \sum_{i=1}^{m} D_t(i)[y_i \neq h_j(x_i)] \tag{11}$$

with D_t being the importance weight for each sample i, with value x_i and label y_i, at each iteration t. D_t is given by

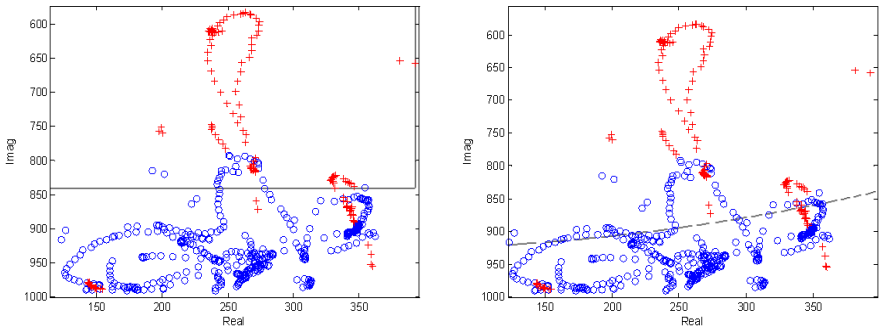
$$D_t(i) = \frac{D_{t-1}(i) \cdot e^{-\alpha_t y_i h_{t-1}(x_i)}}{Z_{t-1}} \tag{12}$$

where Z_{t-1} is a normalization factor chosen so that D_t is a distribution.

There exist many methods in which this decision can be calculated, one such optimised and fast approach [30] computes cumulative histograms per feature for each of the classes. The histograms allow for the selection of a thresholding bin, chosen to maximise the number of samples of one class whilst minimising the number of the other. The point of minimum error is obtained and for each iteration step of the Adaboost algorithm the feature with the lowest minimum error is selected as the weak classifier.

This concept forms the foundation of the proposed method. Cumulative histograms per feature are modelled as bi-dimensional distributions allowing for the use of complex numbers. The use of complex number theory extends the interpretation of a linear one dimensional space into two. Within this space, any given complex number $re + im \cdot i$ is now represented as a point (re, im). This alters the mathematical meaning and significance of concepts such as minimum and maximum, thus altering the actual definition and implementation of the weak classifiers.

As before a threshold point is obtained, that takes into account that the max and min operators have a different interpretation in the complex number space. The threshold is used as a linear decision border by applying the operators to the real and imaginary parts, or as a curve border by applying it to the magnitude and angle (Figure 8). In the same way the complex number space



(a) (b)

Figure 8: An example of the robust 3D VHOG descriptor is shown with the decision border calculated by the first weak classifier on the complex space considering (a) a linear border for the cartesian space or (b) a curved one for the polar space. In our evaluation process the linear case was selected experimentally.

can also be reinterpreted as polar coordinates rather than cartesian, by using the real and imaginary coordinates as module and phase prior to the creation of the bi-dimensional histograms (Figure 9e).

With either case, it is important to outline the differences that the proposed methodology has as opposed to using conventional Adaboost with the real and imaginary parts as independent features. In essence using conventional Adaboost in this way would not respect the complex number nature of the feature source. The relationship between the imaginary and real numbers is not independent but interrelated as a result of the complex number phenomenon. Thus by considering them in isolation that link is lost, this leads to a less rich decision as only half of the information is available when the optimisation search is applied.

To preserve this link; the optimization search to find the threshold, which provides the minimum error in the feature space, is extended from one dimension into a two dimensional search. This however increases computational time, to avoid this an efficient use of feature data is integrated into the methodology, which requires fewer iterations. The cumulative distributions are calculated

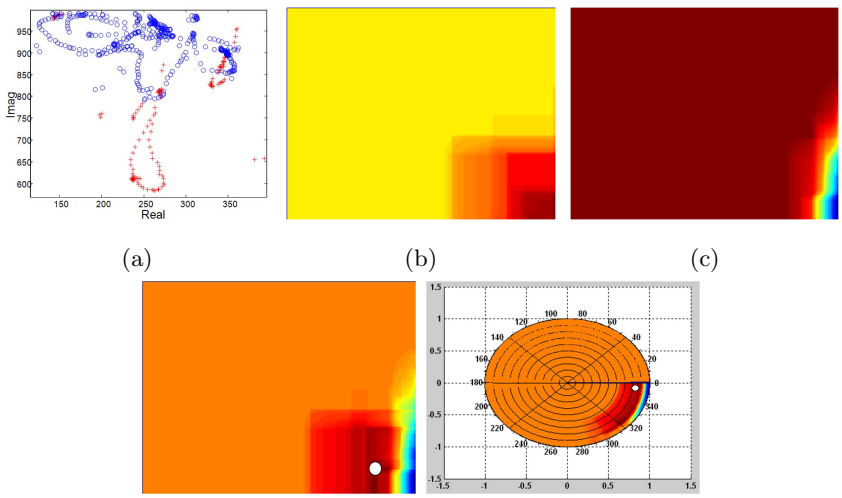


Figure 9: Example of weak classifier for a complex feature. a) Samples for two classes represented in the complex domain (y-axis flipped to fit with the integral images). b) Cumulative distribution of class 1 samples calculated using integral image and with range of colours moving from red (minimum) to blue (maximum) corresponding to the summed areas. c) Cumulative distribution of class 2 samples. d) error distribution and point of minimum error (white spot) calculated by the complex weight classifier. e) error distribution and point of minimum error (white spot) understanding the complex space as polar coordinates.

by applying the integral image [51, 4]. Instead of evaluating each possible hypothesis until finding the optimum, leading to the consequential computational repetition of overlapping areas, a cumulative distribution function is precalculated (Figure 9). The application of the integral image technique allows us, in a single pass over the distribution, to efficiently compute a bi-dimensional cumulative distribution function using the following equation:

$$Q(f,c) = Q(f,c-1) + Q(f-1,c) - Q(f-1,f-1) + h(f,c) \quad (13)$$

where h is the original distribution function, modelled as a histogram. Q is the cumulative integral image and f and c are the column and row indexes, respectively.

Figure 10: Some objects of the new 3D Risk Scenes (3DRS) dataset.

In a similar manner that complex numbers extend the feature space to a two dimensional space, quaternions extend it to a four dimensional space (and to three dimensions in case of pure quaternions). As such the proposed methodology is extendable to higher numbers of dimensions, importantly without assuming independence between the values of these vectors and therefore without losing any of the relational information.

To allow for this, and in the case of quaternions, the optimisation search step must be done in a four dimensional space to find the decision threshold. By replacing the integral image with a multidimensional extension of the integral image [48, 26], the required four dimensional cumulative histogram can be efficiently calculated and the threshold can be extracted. Therefore (13) is transformed to:

$$Q_D im = \sum_{p \in \{0,1\}^d} (-1)^{d-\|p\|_1} Q(x^p) \tag{14}$$

where d is the image dimension, Q is the bi-dimensional integral image of the histogram h, and x^p represents the multidimensional rectangle $[x_0, x_1]$ to be evaluated at each position.

Finally, multi-Adaboost is applied using the one-against-one approach by constructing several binary classifiers for each pair of classes and training over the instances from both classes. In order to obtain the final classification, the individual results are combined using a majority vote.

3.6. Overall risk score estimation

An overall risk score for each scene is finally calculated combining the previous equations for Stability (5) and Hazard Features (10), based on (2). These values are normalised and the weights w_S and w_H can be selected based on the expected application. For example in a chemistry lab, the weighting given to the stability of objects would be higher than to the presence of hazardous objects. This would add more credence to the presence of containers in unstable positions rather than hazardous objects within the environment. The proposed framework can be extended to support any other forms of measurable risk (e.g. temperature) through the addition of extra terms in (2) based on (1). Therefore the risk analysis system can be tailored to each individual environment (e.g. chemistry lab, smart home, etc.) based on circumstance (e.g adults, at risk persons) and the available acquisition devices. Importantly the framework requires no temporal knowledge to estimate the risk as such they system runs on a per frame basis. However due to the computation requirements of the preprocessing and complexity of feature extraction it is currently not a online implementation.

4. Results

The following section outline the evaluation process used to assess the viability of the proposed methodology. Initially an overview of the dataset and evaluation environment is given, followed by individual sections that relate to separate aspects of the proposed methodology.

4.1. Evaluation process

To effectively test the proposed methodologies we make use of the 3D Risk Scenes (3DRS) dataset. Using the Microsoft's Kinect and Kinect Fusion [25], 27 real objects (Figure 10) and 42 indoor scenes (Figure 11 and 12) were captured. Additionally, meta data concerning the objects has also been captured manually, providing physical properties such as weight. For the following experiments only

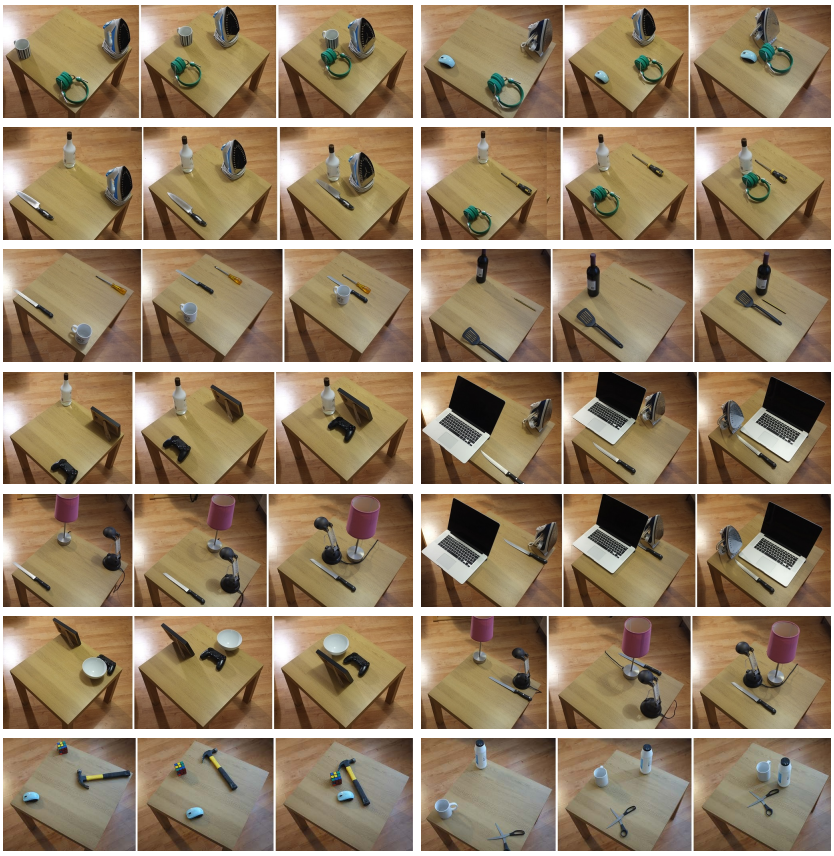


Figure 11: Some scenes of the new 3D Risk Scenes (3DRS) dataset with the three levels of stability for each one.

the RGBD data has been used, the meta data for these objects has not been utilised unless otherwise stated.

Of the 27 objects captured 12 are classified as hazardous with the remaining 15 safe. These include everyday tools and objects commonly found around the home such as knives, irons, balls, cutlery, mugs, bowls, bottles, computer equipment, scissors, vases, etc. Using these objects 42 scenes containing three objects placed on a surface were captured. All scenes were configured on a square table consisting of 3 objects per scene. In more detail, these 42 scenes are split into 14 different scenarios. Each scenario has 3 iterations that represent a different stability level based on the objects predefined locations, i.e. the objects

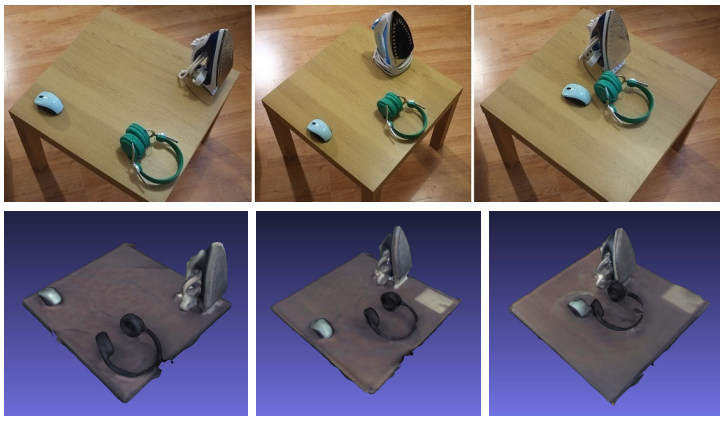


Figure 12: A scene from the new 3DRS dataset reconstructed using Kinect Fusion for the three levels of stability.

are moved closer together on the plane within the scene (Figure 13).

In order to obtain the ground truth for each scene and to ensure that the parameters of the tests are fully controllable, the objects were manually placed on a surface at predefined locations. Each location as we can see in Figure 13, is represented by a different colour which corresponds to a specific stability-risk level.

Each scene and each of the 27 objects are run though the pre-processing step. For all cases a voxel volume representation is returned with a resolution of $256 \times 256 \times 256$ voxels, representing an approximate volume of $50cm^3$. Any lower resolution and shape information about the object would be lost. Additionally, the returned 3D reconstruction of a scene from Kinect Fusion has some preliminary smoothing and hole filling techniques applied, and therefore any higher resolution would not affect significantly the overall performance. The resolution also has a direct impact on computation time for each stage and as such this represents a reasonable trade off for processing time against object detail.

Scene segmentation is part of the pre-processing stage and as such a number of tests were carried out to ascertain the most effective segmentation algorithm to use with the dataset. The segmentation algorithms evaluated included; K-means using a random preliminary clustering phase, Mean Shift with a band-

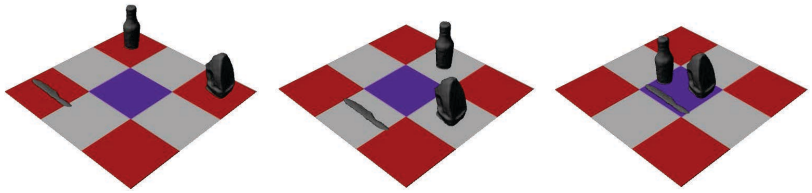


Figure 13: An example scenario with each of its iterations. The level of complexity and stability is increased from (left) a simple layout with lower complexity but higher instability, (mid) to an average complexity and instability, (right) to a complex with lower instability.

width parameter found experimentally, and Distance based clustering based on predefined centroids. Ground truth was established manually and accuracy is defined as the percentage of voxels correctly assigned to their respective object cluster. The results of which are presented in Table 1. As the objects in experiment environment do not touch, the object clusters are defined well enough that a predefined number of clusters is not required to achieve good segmentation. In the instances where voxels are assigned to the wrong object cluster, bounding shapes are still obtained based on the wrongful classification. However, due to the recursive reduction phase, the bounding shapes are iteratively reduced to a point where there is no longer any interaction between them.

The algorithms are evaluated on all scenarios and results are grouped according to stability level, which represents the increasing level of difficulty for the segmentation in each scenario and the reducing instability (Figure 13). Level 1 represents the objects placed at the maximum distance apart, with level three representing all three objects in close proximity. The k-means algorithm was found to be the most efficient at separating the objects across all the complexity-instability levels.

4.2. Stability results

To demonstrate the efficiency of the proposed stability concept, initially 3 experiments were conducted in which an example bounding shape is passed to the physics simulation and the resultant stability was visualised, (Figure 14). The simulation software employed is based on the Bullet 3D Real-Time Multi-

Table 1: Segmentation accuracy for all the levels of stability (see Figure 13 with 1-left, 2-mid and 3-right). Accuracy defined as the percentage of voxels assigned to the correct object cluster.

Stability level	K-Means	Mean Shift	Distance
1	98.86%	97.58%	86.45%
2	86.26%	86.88%	83.32%
3	82.87%	81.62%	78.17%
Overall	**89.33%**	88.69%	82.65%

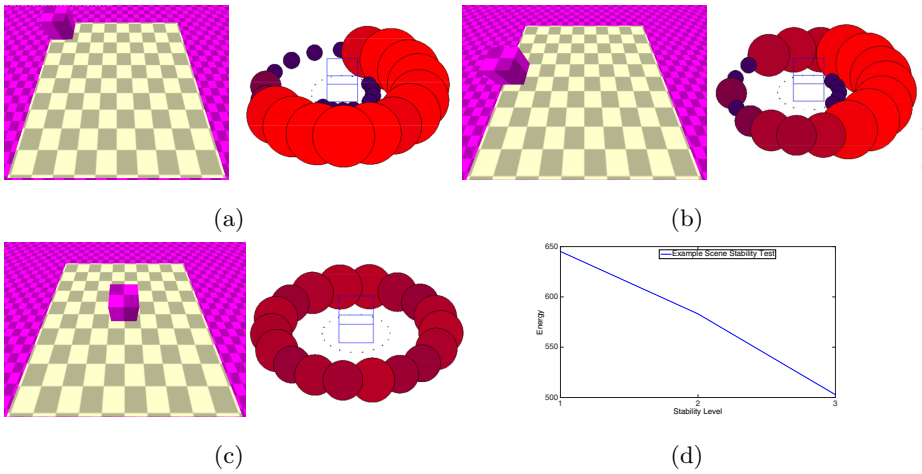


Figure 14: Example scene stability test. Results visualised using circles placed around the object indicating the direction and the level of instability in case (a) Far Left Corner, (b) Left Side and (c) Centered and (d) Scene energy per stability level in graph form. The larger the sphere the more energy output as a result of the force. Additionally emphasized by colouring, where red is a high energy output and blue a low.

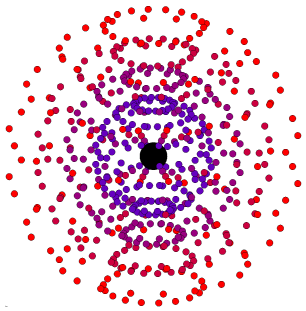


Figure 15: Visual representation of applied forces, force is only applied if the conditions in (4) are met. Each sphere represents the angle from which a force is applied, the distance from the center black sphere represents the magnitude of the applied force.

physics Library [37]. The velocity and angular velocity information for each object at each time frame is extracted and recorded. To visualise the data we position spheres to represent the source (direction) of the force and their magnitude, the further away from an object a sphere is the larger the magnitude of force it represents. The colour and size of each sphere represent the resultant instability, the larger and more red a sphere the higher the energy output as a result of the force applied from that direction. In these examples force was applied from 18 points around the object, each with two levels of magnitude. Forces applied from a direction that would push the object off the table result in the largest energy output, thus represent higher instability.

As with the 3DRS dataset, this example scene has three levels of stability. As the object comes towards the centre of the scene we can see that the energy output decreases (Figure 14d). This follows the logical assumption that objects at the centre of a table are less risky than those at the edge or corner.

To further evaluate this, the stability of 42 real scenes from the 3DRS dataset (14 scenarios each with 3 stability levels) were also analysed. For these experiments, force was applied from points (directions), uniformly sampled along a sphere, with various levels of magnitude (Figure 15). As each scene contains more that one object, and all objects in a scene are represented in a simulation, the effect of collisions between the objects is also taken into account. This is

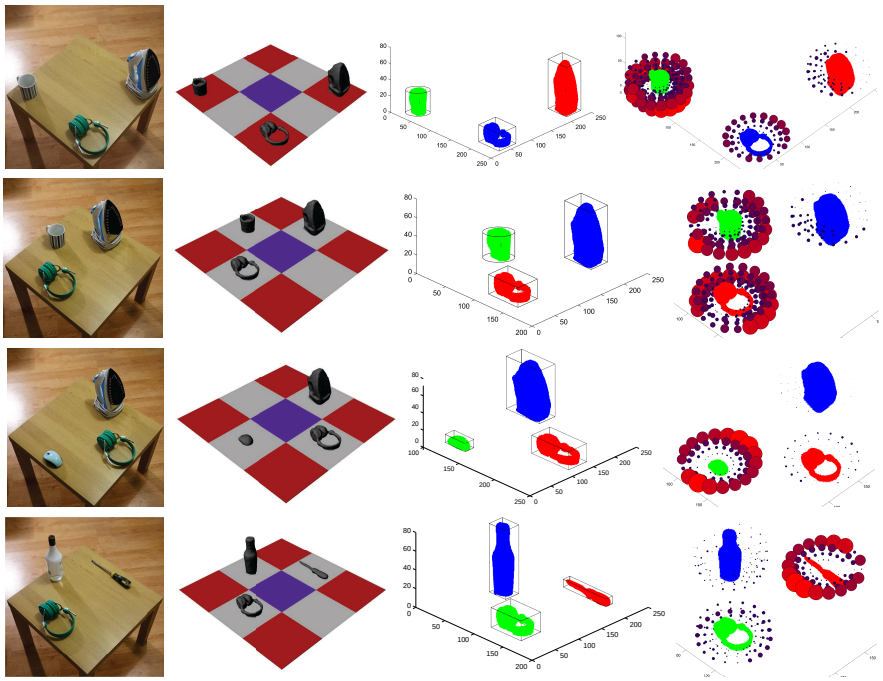


Figure 16: A selection of four scenes with the stability outputs.

visible on the stability plots, especially those of the small objects such as the knife or mouse. For the simulation an object's friction coefficient was globally set 1, while the angular dampening coefficient was experimentally selected to be 0.4. As all objects in the dataset were assigned the same values there is little difference to the results if changed, as such the values chosen have been done so to produce realistic movement for all objects in the dataset in the simulation and according to the suggested values of the physics engine. To maintain an autonomous system a rudimentary measure of mass is given by the number of voxels that each object cluster contains. The scenes' overall stability was quantified according to (3), (4), and (5).

In Figure 16 example estimated stability results are shown. Regarding the collision shapes, three basic primitives can be used; cube, sphere, and cylinder. The most appropriate one can be estimated by simply applying all of them and selecting the one with the least non-object voxels included. The first column of

Figure 16 shows some of the real test scenes, the second contains the outcome of the preprocessing stage, the third shows the scene segmentation results and the obtained bounding boxes and in the last, the Stability Plots with spheres around the objects indicating, with their location, the possible direction of instability and, with their radius/size, the instability level.

Furthermore, in order to compare the proposed stability estimation approach with the current state of the art [57], both methods were tested on the same scenes and the results indicate that the proposed method, which takes into account the possibility that objects may collide with each other, results in more realistic estimates, which are closer to the ground truth. In Table 2 the obtained average stability values for the evaluated 42 scenes are given both for the proposed method and the work presented in [57]. Each scenario becomes more compact and centralised as the stability level changes. Observing the results, it can be seen that as the objects group closer together and move towards the centre of the table the risk score is reduced (Figure 17) in comparison with the work in [57] that has the opposite or no effect. This follows the logical assumption that those items in the center of a table are more stable than those at the edge. It can also be observed, from the stability plots, that additional stability is gained as objects are placed in close proximity to one another, since their potential collisions will reduce the overall instability. It can be observed that the increase in stability is not always uniform, this is in part down to the differing objects in each scene. The properties of the objects, such as size, mass, and shape of the objects will all have an impact on how the stability of a scene changes. For example, a scene with a one larger object and two smaller, will have a distinctly different stability plot to one where the objects are of a more uniform size and mass. This is in part down to the stabilizing effect the larger object would have on the smaller.

4.3. Evaluation of the robust kernel for the 3D shape descriptors

To evaluate the proposed Physics Behaviour Feature (PBF), analysis was conducted on the 27 objects from the 3DRS dataset. Once preprocessed, each

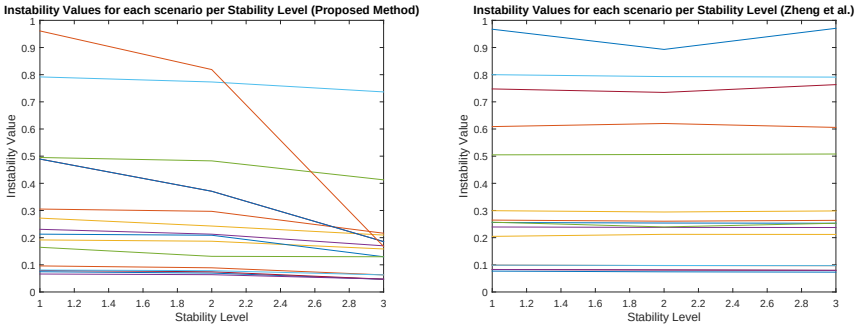


Figure 17: Instability values for each scenario per Stability Level (a) Proposed method, (b) Work presented in [57]. Each line corresponds to one of the 12 scenarios used in our experiments. The vertical axis indicates the stability value obtained using (5), and the horizontal axis indicates the three different stability levels shown in Figure 13. Each of the lines corresponds to one of the scenes. Higher the instability value the less stable the scene is.

Table 2: Average Instability values over all the scenarios at each stability level for the proposed method and the work presented in [57].

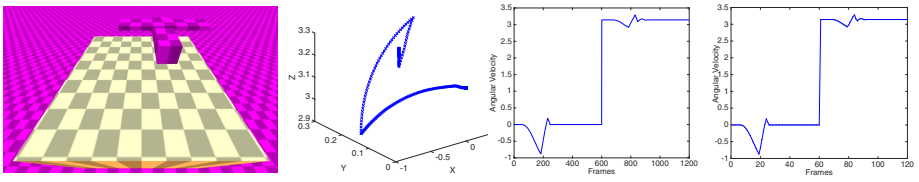
Method	Lv1	Lv2	Lv3
Proposed (Mean)	0.3469	0.3138	0.2199
Proposed (STD)	(0.1242)	(0.1041)	(0.0535)
Zheng [57] (Mean)	0.3837	0.3766	0.3833
Zheng [57] (STD)	(0.1329)	(0.1238)	(0.1339)

object and its resultant bounding shape information was used to perform physics simulations. In order to improve the accuracy of the simulations customised bounding shapes that best suit the objects can be used and mass information is supplied for each object in the 3DRS dataset.

Several features were investigated and evaluations were carried out to establish which one is the most suitable. Due to the nature of the data and that are represented in three dimensions, initially all the available components were utilized to create a feature vector. A pure quaternion representation was considered where the use of the x, y and z values make up the imaginary components. Regarding the x, y and z values represent location in 3D space, velocity or the

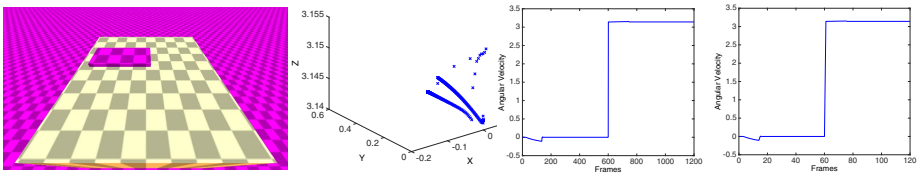
angular velocity. Experimentation was also carried out by reducing the initial data down to two dimensions combined in a complex representation. Furthermore PCA was used to identify if other projections could be more suitable. In all the evaluated cases, the features were tested with and without the proposed complex (or hyper-complex) representation. Both of these complex forms compliment the use of the Complex and Hyper complex Adaboost, allowing the exploitation of the relationships between the dimensions of the data to be taken into account. In this case it was found that the most suitable form was utilising just the x and z components of the angular velocity.

A visualisation of this feature selection process can be seen in Figure 18-19 for two different objects. Subfigure (a) shows the collision shapes in the simulation, (b) the 3 components (x,y,z) of the angular velocity plotted over time, (c) the dimensionality reduction and (d) the final feature vector after down-sampling.



(a) (b) (c) (d)

Figure 18: Physics Feature extraction before and after the dimensionality reduction and the down-sampling stages.



(a) (b) (c) (d)

Figure 19: Physics Feature extraction before and after the dimensionality reduction and the down-sampling stages.

About the other 3D shape descriptors, the 3D HOG is based on the work in

[7], 3D Voxel HOG was based on the work in [18], the 3D SIFT implementation based on the papers [41, 23], the 3D Harris implementation considers the work in [44] and finally the FAST 3D implementation based on the work in [38].

Additionally to test the effectiveness of the proposed robust kernel, the feature vectors for all of the above descriptors have also the kernel applied, providing a comprehensive review of its performance. For the ground truth we define an object as either dangerous or not. However most of the tested descriptors operate on local areas of the voxel volume, thus ground truth for each of these blocks or feature spaces is also defined. All descriptors were trained with the same training set using both Adaboost [21] and the proposed Complex Adaboost. For testing the 'leave-one out' protocol was used and a set number of iterations (500) was specified to create the models. This number was found experimentally to produce the best overall classification models for the dataset. In some cases convergence would be reached sooner.

Regarding the 3D descriptors (3D Harris, 3D SIFT, FAST 3D, 3D VHOG and 3D HOG) based on experimental results and where relevant the values for block and cell size were set to 2 cubic cells and 16 cubic voxels respectively. Table 3 outlines the results of each 3D feature descriptor on the 3DRS dataset, additionally the improvement gained through the use of the novel robust kernel is also displayed.

It can be seen that many of the well known feature descriptors are applicable to this task. However 3D Harris and FAST 3D both performed poorly, this is in part down to a lack of convergence when training the model, as well as a tendency to over fit and as such do not provide a consistent enough description of this local phenomena potentially due to small variations in the voxels or due to the voxel resolution of the scene. From the average results obtained, the overall F1 score was improved by 7.57% indicating the proposed robust kernel has strong potential for use with most of the well-known 3D descriptors.

When compared with other features, PBF shows promising results in the detection and classification of objects in this approach. The formation of the feature vector has a direct influence on the types of objects that are well clas-

Table 3: Comparison of proposed methodologies versus existing 3D Feature Methods with and without the proposed robust kernel.

Feature	F1	Sensitivity	Precision	Accuracy
3D HOG	0.699	0.750	0.600	0.667
Robust 3D HOG	0.686	1.000	0.522	0.593
3D Voxel HOG	0.714	0.833	0.625	0.704
Robust 3D Voxel HOG	0.769	0.833	0.714	0.778
3D SIFT	0.545	0.500	0.600	0.630
Robust 3D SIFT	0.571	0.667	0.500	0.566
3D Harris	0.267	0.167	0.667	0.593
Robust 3D Harris	0.353	0.250	0.600	0.593
FAST 3D	0.000	0.000	1.000	0.556
Robust FAST 3D	0.261	0.250	0.273	0.370
PBF	0.690	0.833	0.588	0.667
Robust PBF	0.727	0.667	0.800	0.778
PBF+3D VHOG	0.750	1.000	0.600	0.704
Robust PBF+3D VHOG	**0.828**	1.000	0.706	0.815
Average 3D	0.5236	0.5833	0.6686	0.6459
Average Robust 3D	**0.5993**	0.6667	0.5879	0.6419

sified. This property of the feature could be exploited to classify other aspects of an object. A combination of the proposed physics (PBF) and the shape (3D VHOG) was devised. To ensure the safest results the two features were fused using an 'OR' operator on an objects classification as hazardous. If either PBF *or* 3D VHOG returns a result of hazardous then that object is deemed unsafe. This combination of features allows analysis of an object cluster on both a local level (3DVHOG) but also at an overall shape level (PBF). This combined descriptor results an overall improvement as shown in Table 3 indicating that their fusion allows to accurately recognise risky and safe objects.

Precision is the fraction of retrieved instances that are relevant, defined as the

true positive rate divided by the number of correctly identified classifications. Sensitivity is the fraction of relevant instances that are retrieved, defined as true positive rate divided by the number of positive results that should have been classified. Both precision and recall are therefore based on an understanding and measure of relevance. Accuracy is a description of systematic errors, or a measure of statistical bias. Finally the F1 Score is another measure of accuracy that uses precision and sensitivity to compute its score.

These results clearly outline that with the use of the proposed robust kernel, improvements in the F1 score and in most cases the sensitivity can be seen on a wide range of 3D descriptors providing more accurate and robust classifications.

4.4. Performance evaluation of Complex and Hyper Complex Adaboost

To evaluate the advantages of the proposed Complex Adaboost the complex 3D feature vectors obtained after using the proposed kernel were compared with the classic Adaboost in terms of complexity. A comparison is given in terms of the training time and the number of iterations required. As before the maximum number of training iterations was specified to 500. Testing was carried out on an i7-4870 2.5GHz PC with 16GB RAM running Windows 8.

The results in Table 4 were derived from the average results from 27 generated models in each descriptor. The iterations were limited to 500, thus results with this number of iterations did not converge. We can see that computational speed gain is considerable with similar numbers of iterations being completed within a fraction of the time needed with conventional Adaboost.

To outline the advantages of Hyper Complex Adaboost, experiments were conducted on a 3 dimensional permutation of the PBF. 16 different feature vector combinations, utilising all three axis of either the angular velocity, rotational velocity or Position, were analysed using both Adaboost and the proposed Hyper Complex Adaboost. The feature vectors were either concatenated vectors of all the data or Hyper Complex variants where the three axis made up the imaginary components of the hyper complex number. The average results for the 16 experiments is shows in Table 5. As can be expected the results of the

Table 4: Complex Adaboost vs standard Adaboost, training times and iterations comparison.

	Standard Adaboost		Complex Adaboost	
Feature	Time(s)	Avg #Iter.	Time(s)	Avg #Iter.
3D HOG	348.57	103.96	9.08	46.96
Robust 3D HOG	651.46	40.67	45.15	14.82
3D Sift	855.16	72.92	19.95	72.92
Robust 3D Sift	1603.66	61.74	43.77	78.96
3D Harris	2261.00	500	46.268	500
Robust 3D Harris	4576.38	500	106.71	500
Fast 3D	2351.40	500	52.33	500
Robust Fast 3D	15959.70	500	93.88	500
PBF	162.56	4.41	1.44	9.26
Robust PBF	1781.7	4.15	4.04	6.98
Average 3D	1195.74	236.26	**25.81**	**225.83**
Average Robust 3D	4914.58	221.31	**58.71**	**220.15**

hyper complex variant of the feature vector with the standard Adaboost has the lowest average results. Utilising the hyper complex feature vector with the proposed Hyper Complex Adaboost, the highest rate of accuracy is achieved. The overall results are comparatively low and as such the use of all three axis in the final PBF+3DVHOG feature was detrimental to performance. However these results illustrate the advantages of the use of hyper complex features and the proposed Hyper Complex Adaboost.

4.5. Overall Risk Scores

An overall confidence (risk) score for each scene is finally estimated combining the previous partial results using (1), (5) and (10); with all the results shown in table 6. About the ground truth it is available since areas of high, medium and low instability are defined as we can see in Figure 13. The ground truth for the unsafe objects is again given from our database where each object is labeled

Table 5: Hyper Complex (HC) Adaboost vs standard Adaboost accuracy evaluation.

	F1	Sensitivity	Precision	Accuracy
3 Axis Feature w/ Adaboost	0.293	0.276	0.389	0.488
3 Axis Feature w/ HC Adaboost	0.292	0.318	0.292	0.456
HC Feature w/Adaboost	0.108	0.073	0.210	0.472
HC Feature w/HC Adaboost	**0.348**	**0.365**	**0.438**	**0.537**

Table 6: Overall Hazard (shape properties) and Instability scores for the testing objects averaged for all the scenarios in each level with higher values indicating higher risk (e.g. presence of sharp features, close to the corner, etc.).

Risk Score	Level 1	Level 2	Level 3	Error
Instability Proposed	0.1487	0.1387	0.1075	**0.074**
Instability Zheng et al.	0.1643	0.1616	0.1627	0.095
Hazard Features VHOG	0.2500	0.2500	0.2500	0.1944
Hazard Features PBF	0.3056	0.3056	0.3056	0.1389
Hazard Features PBF+VHOG	0.3778	0.3778	0.3778	**0.0667**

as safe or not and this information is then utilised in each scene. Total risk is defined as the weighted sum of the Hazard and Instability scores based on (2) with $w_S = w_H = 0.5$ for all the scenes.

Table 7 outlines the hazard scores of each object of the 3DRS dataset according to the PBF+3DVHOG feature descriptor. It can be seen that in most cases the risk score is high for objects that demonstrate some kind of risk e.g the four types of knives, the irons, hammer and the two sets of scissors. Equally less hazardous items are scored low; the ball, bowl, mug etc. However there are cases where the descriptor has been over sensitive, the rubix cube and laptop being examples of this. In the given scenarios it is important for the descriptor to be over sensitive to risk so as to ensure that no hazards are overlooked.

Additionally a breakdown of the calculated risk score per scene, taking into account both the stability of the objects and their respective hazard features is

Table 7: Risk Score of individual objects calculated using PBF+VHOG feature.

Object	Bal	Bot	Bow	Con	Fra	Ham	Hed	Ir	Ir2
Hazard Score	0.06	0.00	0.03	0.16	0.93	0.78	0.32	0.77	1.00
Object	Kn	Kn2	Kn3	Kn4	Lp	Lp2	Lap	Mse	Mug
Hazard Score	0.86	0.86	0.82	0.82	0.10	0.22	0.76	0.21	0.25
Object	Pnc	Pno	Pen	Rub	Slt	Sc	Sc2	Scr	Spt
Hazard Score	0.02	0.78	0.88	1.00	0.93	0.76	0.76	0.92	0.79

(a) (b) (c)

Figure 20: Illustration of instability per iteration of an example scene. As the objects get closer together and further from the edges of the table the instability score goes down

given in Table 8. As the weighting for each risk element is equal in this case, the effect is that the risk scores are smoothed out over the different iterations. With the adjustment of these scores a system can be designed to better illustrated relevant risk in a given environment.

5. Conclusions

In this work the concept of risk analysis is presented for 3D scenes and novel solutions are introduced by combining computer vision and Newtonian physics. A robust kernel for 3D descriptors and a new approach to evaluate the overall stability of a scene were introduced and tested. Also, due to the

Table 8: Risk score per scene. Using PBF+VHOG feature and Stability estimation

Scene	1	2	3	4	5	6	7	8
Lv1	0.271	0.414	0.395	0.356	0.632	0.923	0.235	0.535
Lv2	0.268	0.409	0.392	0.344	0.623	0.911	0.231	0.533
Lv3	0.253	0.357	0.374	0.317	0.578	0.888	0.214	0.481
Scene	9	10	11	12	13	14	15	16
Lv1	0.250	0.368	0.226	0.504	0.240	0.509	0.650	1.00
Lv2	0.246	0.349	0.224	0.482	0.239	0.432	0.573	0.983
Lv3	0.229	0.328	0.214	0.481	0.229	0.312	0.452	0.704

local nature of the proposed 3D features, issues relating to the normalization of a mesh are avoided, removing a potentially complex pre-processing step. Furthermore, features based on the objects' angular velocity are introduced allowing classification of objects as safe and unsafe. Additionally, a complex version of Adaboost was suggested that can exploit the correlation between the real and imaginary elements of complex descriptors with lower complexity. An extended version of the 3DRS dataset was provided for 3D scene risk analysis and experiments were performed showing that the proposed approach has the potential to accurately measure risks in scenes providing good estimates.

It is the intension of the authors to further develop the Risk Estimation Framework to improve the speed and computational time as well as through the use of additional risk elements, such as human interaction, to enrich the initial risk score of a potential hazard.

6. References

[1] T. Adali, P. Schreier, and L. Scharf, "Complex-valued signal processing: The proper way to deal with impropriety," *IEEE Trans. Signal Processing (overview paper)*, vol. 59, no. 11, p. 51015123, 2011.

[2] P. W. Battaglia, J. B. Hamrick, and J. B. Tenenbaum, "Simulation as

an engine of physical scene understanding." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 110, no. 45, pp. 18 327–32, nov 2013.

[3] A. N. Belbachir, A. Nowakowska, S. Schraml, G. Wiesmann, and R. Sablatnig, "Event-driven feature analysis in a 4D spatiotemporal representation for ambient assisted living," *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pp. 1570–1577, nov 2011.

[4] B.Han, C. Yang, R. Duraiswami, and L. Davis, "Bayesian filtering and integral image for visual tracking," in *WIAMIS*, 2005, pp. 329–336.

[5] D. C. Blanchard, G. Griebel, R. Pobbe, and R. J. Blanchard, "Risk assessment as an evolved threat detection and analysis process." *Neuroscience and Biobehavioral Reviews*, vol. 35, no. 4, pp. 991–8, mar 2011.

[6] N. Buch, S. A. Velastin, and J. Orwell, "A review of computer vision techniques for the analysis of urban traffic," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 3, pp. 920–939, sep 2011.

[7] N. Buch, M. Cracknell, J. Orwell, and S. Velastin, "Vehicle localisation and classification in urban CCTV streams," *16th World Congress and Exhibition on Intelligent Transport Systems and Services*, pp. 1–8, 2009.

[8] D. Casanova, J. Florindo, M. Falvo, and O. Bruno, "Texture analysis using fractal descriptors estimated by the mutual interference of color channels," *Information Sciences*, vol. 346, no. 10, pp. 58–72, feb 2016.

[9] L. Chen, C. D. Nugent, and H. Wang, "A knowledge-driven approach to activity recognition in smart homes," *IEEE Transactions On Knowledge and Data Engineering*, vol. 24, no. 6, pp. 961–974, 2012.

[10] P. Cirujeda, Y. Dicente Cid, X. Mateo, and X. Binefa, "A 3D scene registration method via covariance descriptors and an evolutionary stable strategy game theory solver: fusing photometric and shape-based features," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 306–329, 2015.

[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings - IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. I, 2005, pp. 886–893.

[12] T. Dannenmann, "Novel safety feature to protect critical anatomical structures during navigation-guided robotic surgery," in *Jahrestagung der Deutschen Gesellschaft fur Computer-und Roboterassistierte Chirurgie, CURAC 2*, 2003.

[13] A. Davis, K. L. Bouman, J. G. Chen, M. Rubinstein, F. Durand, and W. T. Freeman, "Visual vibrometry: estimating material properties from small motion in video," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5335–5343.

[14] B. Dhillon, A. Fashandi, and K. Liu, "Robot systems reliability and safety: a review," *Journal of Quality in Maintenance Engineering*, vol. 8, no. 3, pp. 170–212, 2002.

[15] C. Do and B. Javidi, "3D Integral imaging reconstruction of occluded objects using independent component analysis-based K-means clustering," *Journal of Display Technology*, vol. 6, no. 7, pp. 257–262, 2010.

[16] B. Drost, M. Ulrich, N. Navab, and S. Ilic, "Model globally, match locally: efficient and robust 3D object recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*. Ieee, jun 2010, pp. 998–1005.

[17] R. Dupre, V. Argyriou, D. Greenhill, and G. Tzimiropoulos, "A 3D scene analysis framework and descriptors for risk evaluation," in *International Conference on 3D Vision (3DV)*. IEEE, 2015, pp. 100–108.

[18] R. Dupre, V. Argyriou, G. Tzimiropoulos, and D. Greenhill, "3D Voxel HOG and risk estimation," in *IEEE International Conference on Digital Signal Processing*, 2015, pp. 482–486.

[19] A. J. Fitch, A. Kadyrov, W. J. Christmas, and J. Kittler, "Fast robust correlation," *IEEE Transactions on Image Processing*, vol. 14, no. 8, pp. 1063–1073, 2005.

[20] A. Flint, A. Dick, and A. Van Den Hengel, "Thrift: Local 3D structure recognition," in *Digital Image Computing Techniques and Applications: 9th Biennial Conference of the Australian Pattern Recognition Society*, 2007, pp. 182–188.

[21] Y. Freund and R. Schapire, "A Desicion-theoretic generalization of on-line learning and an application to boosting," in *Computational learning theory*, 1995, pp. 23–37.

[22] A. Frome, D. Huber, R. Kolluri, T. Bülow, and J. Malik, "Recognizing objects in range data using regional point descriptors," in *ECCV*, 2004, pp. 224–237.

[23] A. Godil and A. Wagan, "Salient local 3D features for 3D shape retrieval," in *IS&T/SPIE Electronic Imaging*, 2011, pp. 78 640S—-78 640S.

[24] J. Huang, R. Yagel, V. Filippov, and Y. Kurzion, "An accurate method for voxelizing polygon meshes," in *IEEE Symposium on Volume Visualization*. Ieee, 1998, pp. 119–126.

[25] S. Izadi, A. Davison, A. Fitzgibbon, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, and D. Freeman, "Kinect Fusion: Real-time 3D reconstruction and Interaction ising a moving depth camera," in *Proceedings of the 24th annual ACM symposium on User Interface Software and Technology*, 2011, pp. 559–568.

[26] Y. Ke, R. Sukthankar, and M. Hebert, "Efficient visual event detection using volumetric features," in *IEEE Conference on Computer Vision*, vol. 1, 2005, pp. 166–173.

[27] B.-S. Kim, P. Kohli, and S. Savarese, "3D Scene understanding by voxel-CRF," in *IEEE International Conference on Computer Vision*. Ieee, dec 2013, pp. 1425–1432.

[28] A. Kläser, M. Marszałek, C. Schmid, and A. Zisserman, "Human focused action localization in video," in *Trends and Topics in Computer Vision*, vol. 6553 LNCS, 2012, pp. 219–233.

[29] Y. Kobayashi, T. Morimoto, I. Sato, Y. Mukaigawa, and K. Ikeuchi, "BRDF Estimation of structural color object by using hyper spectral image," in *IEEE International Conference on Computer Vision Workshop*, dec 2013, pp. 915–922.

[30] N. Lawrence, "Gaussian process latent variable models for visualisation of high dimensional data," *Advances in Neural Information Processing Systems*, vol. 16, no. 3, pp. 329–336, 2004.

[31] X. Li, T. Adali, and M. Anderson, "Noncircular principal component analysis and its application to model selection," *IEEE Trans. Signal Processing*, vol. 59, no. 10, p. 45164528i, 2011.

[32] D. Lowe, "Object recognition from local scale-invariant features," in *IEEE Conference on Computer Vision*, vol. 2. Ieee, 1999, pp. 1150–1157.

[33] Z. C. Marton, D. Pangercic, N. Blodow, J. Kleinehellefort, and M. Beetz, "General 3D modelling of novel objects from a single view," in *IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems, IROS 2010 - Conference Proceedings*, 2010, pp. 3700–3705.

[34] New Zealand. Department of Labour. Industrial Welfare Division., *Robot Safety*. Industrial Welfare Division, Department of Labour, 1987.

[35] J. Niemeyer, F. Rottensteiner, and U. Soergel, "Contextual classification of lidar data and building object detection in urban areas," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 87, no. 1, pp. 152–165, 2014.

[36] A. Prest, V. Ferrari, and C. Schmid, "Explicit modeling of human-object interactions in realistic videos." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 4, pp. 835–48, apr 2013.

[37] Real-Time Physics Simulation, "Bullet User Manual and API Documentation," 2012.

[38] E. Rosten, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 105–119, 2010.

[39] R. B. Rusu, N. Blodow, and M. Beetz, "Fast Point Feature Histograms (FPFH) for 3D registration," in *IEEE International Conference on Robotics and Automation*, 2009, pp. 3212–3217.

[40] M. Scherer, M. Walter, and T. Schreck, "Histograms of oriented gradients for 3d object retrieval," in *International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, 2010, pp. 41–48.

[41] P. Scovanner, S. Ali, and M. Shah, "A 3-dimensional sift descriptor and its application to action recognition," in *Proceedings International Conference on Multimedia*. New York, New York, USA: ACM Press, 2007, pp. 357–360.

[42] C. Sharp, O. Shakernia, and S. Sastry, "A vision system for landing an unmanned aerial vehicle," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 2. Ieee, 2001, pp. 1720–1727.

[43] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *IEEE Conference on Computer Vision and Pattern Recognition*. Ieee, jun 2011, pp. 1297–1304.

[44] I. Sipiran and B. Bustos, "Harris 3D: a robust extension of the Harris operator for interest point detection on 3D meshes," *The Visual Computer*, vol. 27, no. 11, pp. 963–976, jul 2011.

[45] E. Stone and M. Skubic, "Evaluation of an Inexpensive Depth Camera for Passive In-Home Fall Risk Assessment," in *Proceedings of International ICST Conference on Pervasive Computing Technologies for Healthcare*. Ieee, 2011, pp. 71–77.

[46] A. Swadzba, N. Beuter, S. Wachsmuth, and F. Kummert, "Dynamic 3D scene analysis for acquiring articulated scene models," in *IEEE International Conference on Robotics and Automation*. Ieee, may 2010, pp. 134–141.

[47] S. Tang, X. Wang, X. Lv, T. X. Han, J. Keller, Z. He, M. Skubic, and S. Lao, "Histogram of oriented normal vectors for object recognition with a depth sensor," in *Proceedings of Asian conference on Computer Vision*, vol. 2, 2012, pp. 525–538.

[48] E. Tapia, "A note on the computation of high-dimensional integral images," *Pattern Recognition Letters*, vol. 32, no. 2, pp. 197–201, 2011.

[49] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *European Conference Computer Vision*, 2010, pp. 356–369.

[50] A. Trevor, S. Gedikli, R. B. Rusu, and H. I. Christensen, "Efficient organized point cloud segmentation with connected components," in *Proceedings of Semantic Perception Mapping and Exploration*, 2013, pp. 1–6.

[51] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE Conference on Computer Vision and Pattern Recognition*, dec 2001, pp. 329–336.

[52] J. Wang, C. Zhang, W. Zhu, Z. Zhang, Z. Xiong, and P. A. Chou, "3D scene reconstruction by multiple structured-light based commodity depth cameras," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2012, pp. 5429–5432.

[53] O. Wang, P. Gunawardane, S. Scher, and J. Davis, "Material classification using BRDF slices," in *IEEE Conference on Computer Vision and Pattern Recognition*, jun 2009, pp. 2805–2811.

[54] M. Weinmann, B. Jutzi, S. Hinz, and C. Mallet, "Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 105, no. 7, pp. 286–304, 2015.

[55] K. E. Wenzel, A. Masselli, and A. Zell, "Automatic Take Off, Tracking and Landing of a Miniature UAV on a Moving Carrier Vehicle," *Journal of Intelligent & Robotic Systems*, vol. 61, no. 1-4, pp. 221–238, oct 2010.

[56] J. Wu, I. Yildirim, J. Lim, W. Freeman, and J. Tenenbaum, "Galileo : Perceiving Physical Object Properties by Integrating a Physics Engine with Deep Learning," in *Advances in Neural Information Processing Systems*, 2015, pp. 1–9.

[57] B. Zheng, Y. Zhao, and J. Yu, "Detecting potential falling objects by inferring human action and natural disturbance," in *IEEE International Conference on Robotics*, 2014, pp. 127–135.

[58] B. Zheng, Y. Zhao, J. Yu, K. Ikeuchi, and S.-C. Zhu, "Scene understanding by reasoning stability and safety," *International Journal of Computer Vision*, vol. 112, no. 2, pp. 221–238, 2015.