AI-based Density Recognition

Simone Müller
Leibniz Supercomputing Centre (LRZ)
simone.mueller@lrz.de

Daniel Kolb
Leibniz Supercomputing Centre (LRZ)
daniel.kolb@lrz.de

Matthias Müller
German Aerospace Center (DLR)
matthias.mueller@dlr.de

Dieter Kranzlmüller
Ludwig-Maximilians-Universität (LMU)
kranzlmueller@ifi.lmu.de

Abstract

Learning-based analysis of images is commonly used in the fields of mobility and robotics for safe environmental motion and interaction. This requires not only object recognition but also the assignment of certain properties to them. With the help of this information, causally related actions can be adapted to different circumstances. Such logical interactions can be optimized by recognizing object-assigned properties. Density as a physical property offers the possibility to recognize how heavy an object is, which material it is made of, which forces are at work, and consequently which influence it has on its environment. Our approach introduces an AI-based concept for assigning physical properties to objects through the use of associated images. Based on synthesized data, we derive specific patterns from 2D images using a neural network to extract further information such as volume, material, or density. Accordingly, we discuss the possibilities of property-based feature extraction to improve causally related logics.

Keywords

AI, Density Recognition, Computer Vision

1 INTRODUCTION

Modern machines and robots use various sensors to capture and navigate their surroundings. Particularly in road traffic, situations may appear inconspicuous at first sight but require constant attention and quick reactions. This can involve evaluating the potential risks in autonomous driving scenarios when a car not only recognizes objects but can also estimate the potential damage in the event of a collision and adapt its driving behavior accordingly. Additional information could increase a system's scope of action and decision-making as well as the automatic assessment of real-life scenes.

Whether it is a ball that rolls onto the road, a car that suddenly brakes, or an item that falls off a moving vehicle. Such reactions are often based on causal relationships that are logical for us humans but not for machines. Despite their logic, machines lack the necessary background knowledge and specific skills, such as the assessment of physical properties, to gain a causal understanding.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Material recognition and the association of related properties can be helpful in causal decision-making [26]. For example, an industrial robot can apply the optimum force for gripping an inelastic object if it knows the approximate material, mass, roughness, and size of this object. All this information relates to the physical density and material property.

Machine learning offers solutions for material recognition [26]. Databases such as Flickr [14] are able to recognize different materials, as shown in Tab. 1.

Table 1: **State-of-the-Art Material Database [14].** Illustration of ten example material categories of Flickr database. The lighting conditions, compositions, colors, textures, surface shapes, material subtypes, and object associations were considered by an image diversity of 100 pictures, 50 close-ups, and 50 normal views in each category [1].

These patterns are recognized by visual features and stored in the model through a learning process, using sophisticated AI algorithms for the recognition of

objects in 2D images, such as YOLOv3 [21], Faster Region-Based Convolutional Neural Networks (R-CNN) [22], and Multi-Scale Convolutional Neural Network (MSCNN) [3]. However, the previously trained 2D object recognition is limited to specific object classes and visual interference effects of images [17]. To collect ambient information and make accurate decisions with high confidence, the AI usually needs to be trained extensively with vast sets of data. Additionally, this material information has usually no connection with the associated physical properties.

Based on the challenges of accurate processing and linking causal information, we present an approach that enables the assignment of physical properties in objects based on a 2D image by using machine learning and pattern recognition. The object is extracted and scaled into triangles to estimate the volume. We derive the associated materials from a database and ultimately calculate the density as a physical quality.

AI-based recognition of density and volume provides a solid foundation for the extraction of additional information from the environment. As an example, object-related forces can be calculated based on equation-specific coefficients, constants, and acquired sensor data including density. This expands the information content in a visual scene. Especially in road traffic, this can be an additional aid to improve the perception of autonomous vehicles.

This paper describes a proof of concept for the implementation of AI-based density recognition. Our work comprises the following contributions:

- Neural-specific object and texture detection based on object classification
- Concept of AI-based density recognition
- Analysis of recognized object density and material composition

Our evaluation reveals the feasibility and transferability of AI-based density recognition. For our empirical examination, we use synthetically generated data from the Unreal Engine.

The paper is organized according to a fixed structure consisting of related work, concept, methodology of AI-based object and texture detection, evaluation, conclusion, as well as future work.

2 MODERN RECOGNITION

This section presents recognition models and existing approaches for physical property recognition. The basic idea involves material recognition, which gives rise to entire databases such as Flickr [14] which assigns materials based on visual appearance.

Liu et al. [14] describe that the visual appearance of a surface depends on illumination conditions, geometric structure of surfaces at different spatial scales, and reflectance properties. Thereby, the reflectance properties of the surface are often characterized by features with a bidirectional reflectance distribution function (BRDF) [18]. In this context, material recognition can employ the recognition of colors and textures, micro-textures, outline shapes, or reflectance-based features [14] by SIFT algorithms. This algorithm can recognize contours based on corners and edges [13].

Standard k-means algorithms Eq. 1 are used to cluster instances of each feature [14] in order to assign the image-specific materials M to the respective words.

$$J = \sum_{i=1}^{k} \sum_{x_j \in S_i} ||x_j - \mu_i||^2 \quad (1)$$

S_i describes the cluster in Eq. 1, which is determined from data points x_j and centroids μ_i on the basis variance minimization [6] and squared Euclidean distance $||x_j - \mu_i||^2$. The random mean value k is determined in the visual data set m_i, ..., m_k. Each data object is assigned to the cluster with the lowest variance for all $l = [1, ..., k]$, shown in Eq. 2.

$$S_i = \{x_j : ||x_j - m_i||^2 \leq ||x_j - m_l||^2\} \quad (2)$$

Machine learning can be used as a static method to learn continuously and specifically from experiences. Thereby, the training-based data is divided into different classes. The classification S_i permits the mapping of input variables $f_{i,x} : \mathbb{R}^{n_i} \rightarrow \mathbb{R}^{n_{i-1}}$ to discrete output variables O. In this view, the regression $g_{(x)}$ refers to a distinction between more than two categories [15]. Fig. 1 summarises the related components of neural networks.

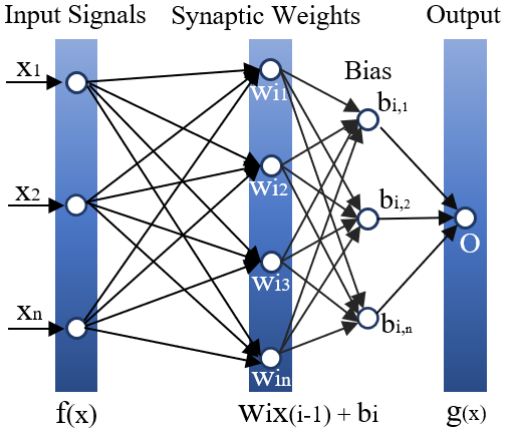

Figure 1: **Schematic Neural Network, adapted from [29].** The input signals $f(i,x)$:[x_1, .. , x_n] are weighted using the weighting factor w_i before they reach the main part of the neuron. A bias b_i is included as a threshold value which must first be exceeded to generate the output signal O.

Maschine learning techniques are broadly divided into two processes: feature extraction and classification. Feature extraction involves the identification and correlation of patterns with large datasets suitable for modeling. Thereby, a feature refers to a property derived from raw data input intending to provide a suitable representation [10]. In both cases, numerous processing nodes of neural networks are tightly interconnected and further layered in organized nodes to perform complex calculations [24].

The overall quality of learning-based feature extraction depends on the task area and associated data set. In this respect, the data must contain a high density of information. To differentiate between classes, the algorithms try to recognize existing patterns of colors, shapes, textures or pixel values within this information [9].

Girshick et al. [8] describe an approach in which high-capacity convolutional neural networks are applied to bottom-up region proposals for localization and segmentation. They introduce a paradigm for training large CNNs when labeled training data is scarce. They detail how pre-training the network with supervision on an auxiliary task with abundant data (image classification) and then fine-tuning the network on the target task where data is scarce (recognition) can improve overall efficiency. This approach is similar to R-CNN. In fast R-CNN, the CNN is first fed with the input image to generate a convolutional feature map. Subsequently, the selective search is performed and the region suggestions are warped into squares. Those region suggestions are called Regions of Interest (RoI) and refer to a subset of the original image [5]. By using RoI pooling layers, each region that has been proposed and which may have different sizes, is reshaped into fixed size so that it can be fed into a fully connected layer. On the output, a softmax layer is used to predict the class of the proposed region and the values of the bounding box [19]. This method produces results faster since it calculates the CNN features only once per image and not two thousand times as with the R-CNN method.

Sean Bell et al. [2] suggest in a direct comparison between three different CNN models (AlexNet, VGG-16, GoogleNet) that material recognition and segmentation of everyday images which are based on Materials in Context Database (MINC) is possible with a probability of 82.2 % (AlexNet) to 85.9 % (GoogleNet).

Shukla et al [25] evaluated the accuracy between CNN classifiers for material recognition and deep learning classifiers. They found that CNN classifiers have better and faster recognition accuracy since the existing probability level allows the classifier to recognize materials with higher accuracy.

Various previous works [3, 16, 21, 22] refer to AI algorithms such as YOLOv3, Faster R-CNN and MSCNN that use a class-verifying diversity of object propositions for object recognition.

Modern Architectures such as YOLO have been continuously improved to perform tasks in the areas of general and oriented recognition, instance segmentation, pose, key points and classification [12]. A DarkNet-19 model architecture (YOLOv2) was expanded to a more complex backbone model DarkNet-53 (YOLOv3) in which features on three different scales can be recognized [4]. Although the implementation of such new functions and targeted optimizations reduce latency times, they often require computationally intensive operations that demand considerable computing power.

Ren et al. [22] found a solution to an issue of R-CNN caused by selective search. Selective search is a rigid algorithm that is unable to improve or learn, which can lead to poor suggestions for candidate regions. They developed the Faster-R-CNN algorithm, replacing selective search with a separate network, the Region Proposal Network (RPN), to predict region proposals. The RPN takes an image as input and outputs a series of rectangular object proposals, each with a class prediction and a confidence value. The network can be trained throughout by backpropagation, where the gradient of the loss function is calculated taking into account the weights of the network for a single input-output example [23].

Wu et al. [28] teach a computational vision system to understand physical relationships with the help of unlabelled videos. They address specific physical scenarios and distinguish between two groups of physical properties: The first inherits the intrinsic physical properties of objects such as volume, material, and mass. The other group is the descriptive physical properties, which describe the scene and are determined by the first group. These include, but are not limited to, the speed of the objects, the distance they travel, or whether they fall into water. The presented model uses CNN to learn the object properties exclusively from unlabelled data. This approach provides serviceable results on a physical data set.

3 DENSITY RECOGNITION

This chapter describes the concept of density and material recognition in order to calculate physical properties like masses. Fig. 2 illustrates the fundamental pipeline of AI-based density recognition.

Building on identifying specific features from 2D images of previous work, we combine object detection with the assignment and calculation of physical properties. The image data is first analyzed by using a neural network. Salient objects can be identified and classified texturally. Object areas are identified as $b_1, ..., b_n$. Within the bounding boxes, possible materials can be

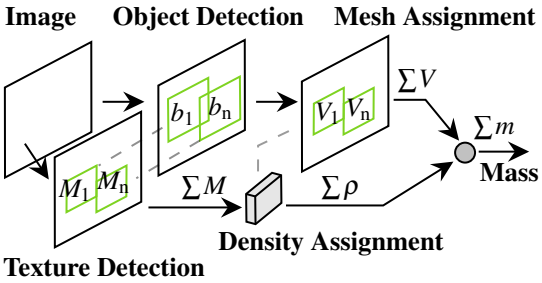

Figure 2: **Pipeline of AI-based density recognition.** The pipeline includes the detection of objects and their textures as well as the assignment of density and meshes to calculate physical quantities like object masses.

attributed based on Flickr Material Database (FMD) and Materials in Context Database (MINC). In this process, these materials M will be assigned a specific literary density ρ where we consider further pattern properties like image colors, shapes, textures and pixels. Since it is necessary to determine the volume V of the respective objects, the process calculates their corresponding meshes. The physical properties like mass M can be derived from the extensive information.

3.1 Object Detection

In order to detect objects, we use the convolutional network of YOLOv4 [12]. We extract features from the input images and calculate them into feature maps. As part of the YOLO architecture, we use backbone, neck, and head detectors, as shown in Fig. 3.

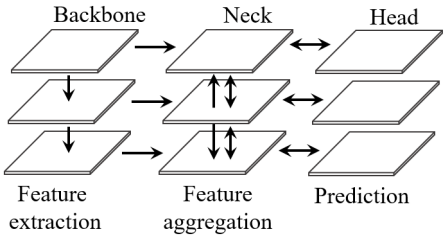

Figure 3: **YOLO Detectors used in this Work.** The backbone extracts important features from the image at different scales. The neck concatenates the semantic information from different layers of the backbone network and transmits it as input to the head. The head applies the refined features for predictive object recognition.

YOLOv4 contains a pre-trained convolutional neural network such as VGG16 or CSPDarkNet53 as a backbone which is based on SPP-Modul (Spatial Pyramid Pooling) and PAN (Path Aggregation Network). As part of the prediction, the head processes aggregated features and predicts the bounding boxes, objecthood, and classification values.

Our model is trained on a MS COCO dataset, which contains over 80 different classes and 1.5 million object instances in 200 thousand labeled images.

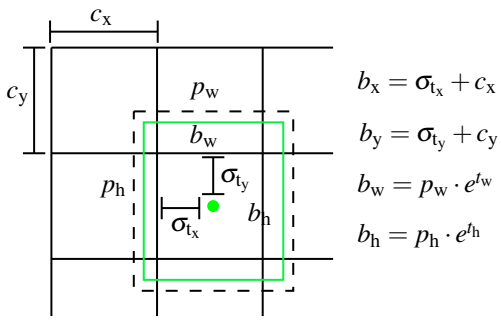

Figure 4: **Bounding Boxes with Dimension Priors and Location Prediction, adapted from [20].** The center coordinates of the box can be calculated with the predicted values t_x, t_y using a sigmoid function and offset by the location of grid cell c_x, c_y. The width and height of the final box are adjusted to the previous width p_w and height p_h and scaled by e^{t_w} and e^{t_h}.

Recognition first divides the image into a grid of cells like shwon in the example in Fig. 4. The number of cells depends on the size of the image. For example, With a size of 608×608 pixels, the cell size is usually 32×32 pixels. Our data set is divided into 19×19 cells.

Each object is assigned to exactly one cell, containing the object's center point. Objectless cells are filtered out according to their low probability of all 80 classes. The use of Non-Max-Suppression, as shown in Fig. 5, eliminates unwanted bounding boxes so that only the most probable bounding box remains for each detected object.

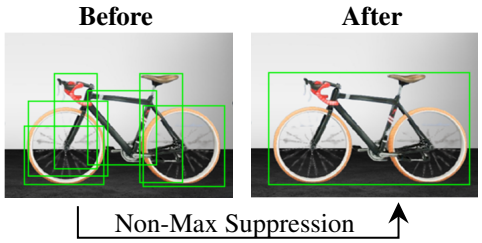

Figure 5: **Effect of Non-Max Suppression (NMS).** The post-processing technique Non-maximum suppression reduces the number of overlapping bounding boxes.

The bounding boxes localize the position of objects in order to recognize possible textures and calculate the object-specific volumes. The area is adjusted to 224×224 pixels for the matching designation of the model input.

3.2 Texture Detection

The object-specific bounding box detection enables the continuous application of a material detection model to the image area of the box. Inside the box, features such as color, SIFT, jet, micro-SIFT, micro-jet, curvature, edge-slice and edge-ribbon are combined and quantized into visual words by Bayesian framework [14].

Three different models are used, consisting of the MINC dataset. MINC comprises 23 different classes, each with 2500 images. We measure the confidence score for all three models across MINC dataset and Flickr Material Database (FMD) for appropriate model selection (see also Tab. 2).

Classes	VGG16	GoogleNet	AlexNet
Fabric	(78 | 78)	(69 | 78)	(45 | 64)
Foliage	(71 | 95)	(68 | 95)	(62 | 93)
Glass	(40 | 82)	(40 | 84)	(27 | 78)
Leather	(29 | 88)	(18 | 84)	(9 | 80)
Metal	(44 | 72)	(37 | 76)	(33 | 69)
Paper	(41 | 90)	(35 | 90)	(11 | 85)
Plastic	(78 | 75)	(84 | 78)	(74 | 68)
Stone	(78 | 89)	(62 | 87)	(52 | 85)
Water	(47 | 96)	(43 | 94)	(30 | 93)
Wood	(36 | 74)	(25 | 78)	(13 | 71)

Table 2: **Recognition on (FMD | MINC)**. VGG16 slightly outperforms GoogleNet. AlexNet provides the lowest accuracy out of these three contrasted models.

FMD consists of ten classes with 100 images of each. Tab. 2 shows the performance of the models for the FMD dataset. VGG16 provides optimum results with an overall accuracy of 86 % at MINC and 52 % at FMD. The model's size, however, requires significant time for both training and recognition. Alternatively, GoogleNet offers a good replacement model with an accuracy of 86 % for MINC and 47 % for FMD. The use of so-called inception modules allows a shorter calculation time. This module replaces a sparse CNN with a normal dense construction since most activations in a deep network are zero values or redundant due to correlations. As a result, not all output channels are connected to the input channels, hence the reduced computing time [27].

3.3 Mesh and Density Assignment

Through the accompanying object recognition, the next step is to assign a suitable 3D mesh. The mesh must have certain properties for the correct calculation of volume. For example, each triangle of the mesh must have corner points stored in a clockwise direction. The mesh must be complete without open areas and completely closed to prevent subsequent miscalculations. Eq. 3 describes the surface calculation of each signed triangle with $V_i \in V$ and $i \in [1,n]$:

$$V_i' = \frac{1}{6}(-x_{i,3}y_{i,2}z_{i,1} + x_{i,2}y_{i,3}z_{i,1} + x_{i,2}y_{i,3}z_{i,1} + x_{i,3}y_{i,1}z_{i,2} - x_{i,1}y_{i,3}z_{i,2} - x_{i,2}y_{i,1}z_{i,3} + x_{i,1}y_{i,2}z_{i,3}) \quad (3)$$

We use i as the index for the triangles. $x_{i,[1,2,3]}$, $y_{i,[1,2,3]}$ and $z_{i,[1,2,3]}$ are the coordinates of the vertices of triangle i. Various shapes of objects are not considered in our analysis. Instead, each class is assigned to a specific 3D mesh. Fig. 6 exemplifies the resulting 3D mesh calculated by Eq. 3.

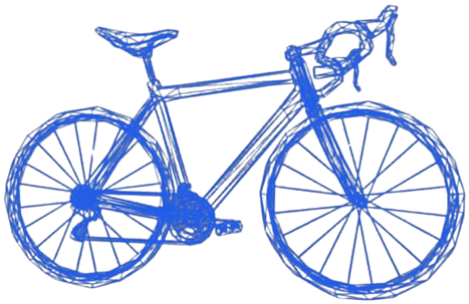

Figure 6: **Object Detection and Triangle Estimation.** Triangles: 5.022, Vertices: 4.159, UV Channels: 4 with approx. Size: 50 × 198 × 114.

After recognition, the applicable model is assigned. The object's mass is attributed based on the result of texture recognition. We simplify our analysis by assuming solid material for the respective model. A further database is created for the classification, which specifies the density of each recognizable material.

3.4 Physical Properties

Physical quantities such as forces, friction, pressures, temperatures, air resistances, inertia moments, energies or material properties n depend on density ρ. The fundamental calculation of density ($\rho = m/V$) and the inclusion of further coefficients and constants offers possibilities of inferring different values.

By recognizing the actual object and the possible material assignment, we can deduce the volume and mass of the object. The physical property is determined by iterating over each section since many objects are divided into sub-objects. The bicycle in Fig. 6, for example, includes the individual wheels, the handlebars, the saddle, and the frame. For each section, the signed volume is now calculated for each triangle and added to the total volume. Surfaces that point outwards contribute to the total volume. Surfaces that point inwards subtract from it. This leaves only the volume on the inside. The density and volume can then be multiplied to calculate the weight of the object.

4 EXPERIMENTAL SETUP

Our approach utilizes Unreal Engine 4.27 and its high-fidelity rendering pipeline. The realistic rendering and lighting allow us to assume real test simulations as shown in Tab. 3. The neuronal training is based on realistic test images.

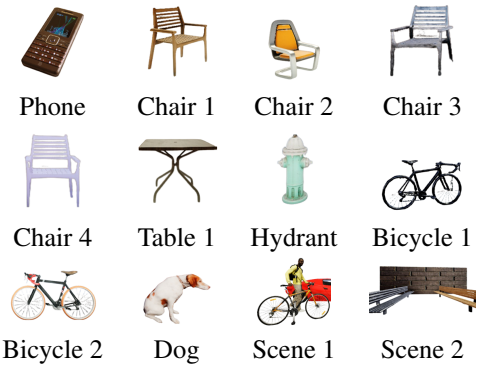

Table 3: **Sample of Items used in our Experiment.** We used Unreal Engine 4.27 for its realistic rendering capabilities.

The accuracy of the physical information is directly linked to the image quality. In our conceptual consideration, we use images with a size of 608×608 pixels and three assigned color channels.

Our computing hardware has an integrated Quad-Core Intel or AMD, 2.5 GHz, 8GB RAM, external GPU1 Nvidia GeForce GTX 1050 Ti, and onboard GPU0 of Intel HD Graphics 630. Cv2 is used for image processing and reading deep neural networks, and NumPy for mathematical functions. Mean subtraction calculates the average pixel intensity over all images of the used training set of all three color channels and subtracts these values from the channels of the input image. When using YOLOv4, channel swapping is also applied for the mean subtraction. Here the image is swapped in RGB order. In order to obtain one optimal bounding box for each object, non-max suppression is applied. To predict the material, we iterate over each object and use the corresponding image area as input to the texture model, similar to the object detection model.

5 EVALUATION

Our evaluation employs density recognition of several diverse objects.

We used a Convolutional Neural Network to recognize the texture of the object and MINC for texture recognition. The model trained and used in our approach shows the highest accuracy when compared to other CNN architectures.

In order to evaluate the functionality of our approach, we explored and tested select scenarios (see Tab. 3). We summarize our results in Tab. 4.

However, the FMD dataset only achieves an accuracy of 52 % across all classes. One of the reasons is the insufficient data set, which consists of only ten categories in the neural network and 23 in the MINC data set. Increasing the data density would significantly increase the hit rate. With more extensive training, our approach

	Material Type	**Density** $[kg/dm^3]$ **(Literary | Measured)**
Phone	Plastic	(4.0 | 1.2) $\pm$ 70 %
Chair 1	Wood	(0.7 | 0.7) $\pm$ 4 %
Chair 2	Plastic	(4.8 | 1.2) $\pm$ 75 %
Chair 3	Metal	(7.9 | 8.0) $\pm$ 1 %
Chair 4	Metal	(0.9 | 1.2) $\pm$ 30 %
Table 1	Metal	(7.9 | 8.0) $\pm$ 1 %
Hydrant	Metal	(7.9 | 8.2) $\pm$ 14 %
Bicycle 1	Metal	(2.9 | 8.0) $\pm$ 180 %
Bicycle 2	Metal	(7.9 | 8.0) $\pm$ 2 %
Bench 1	Metal	(7.9 | 8.0) $\pm$ 1 %
Bench 2	Wood	(2.1 | 0.7) $\pm$ 66 %
Dog	Other	(1.1 | 1.0) $\pm$ 5.6 %
Person	Plastic	(1.1 | 1.0) $\pm$ 9 %
Backback	Fabric	(1.4 | 1.6) $\pm$ 16 %
Car	Plastic	(5.4 | 1.2) $\pm$ 77 %

Table 4: **Detected Materials and Density as well as Percentage Error [%] from the actual Physical Values of the Objects.** The measurements for Phone, Chair 2, Bicycle 1, Bench 2, and Car deviated significantly from the actual values. These deviations stemmed from incorrect detection of the material.

can also be transferred to other environments. This requires the RGB image for analysis and the database with the necessary 3D networks and recognition models for evaluation.

The actual size of objects within a scene has not been considered in previous work. This would be useful for volume calculation and different scaling of objects. Even within the categories, no distinctions are made between different types of objects. Each object is only assigned a 3D mesh, which is considered the average for that class. This means that object shapes are not taken into account. Furthermore, for successful mapping and analysis of physical properties, the use of error-free, detailed, and complete 3D models is essential. However, depending on the orientation and movement of the objects, the calculated volume may be inaccurate. Serial images could help alleviate this error.

6 LIMITATIONS

While our implementation shows promising initial results, it solely serves to illustrate the feasibility of our proposed concept. Although our implementation used neural networks that were trained with real-world images, we relied on synthetic datasets to assess the performance of the implementation. Consequently, the generalizability and applicability of our findings may be limited.

Additionally, our evaluation of the detected physical properties examined the average density of each object.

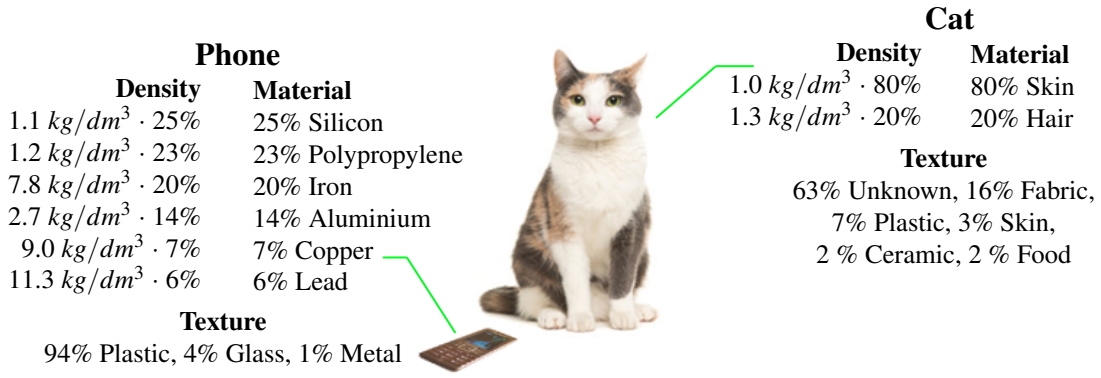

Figure 7: **Material and Density Composition of Recognized Objects.** The inner ingredients are frequently more complex and diverse than the exterior texture suggests. The illustration describes the composition of two example objects: A smartphone and cat. The density of occluded components can be estimated from the average composition of each object.

Therefore, our work might not guarantee that our approach is readily applicable to other properties, such as plasticity or thermal conductivity, as well as to more complex object compositions.

7 CONCLUSION AND FUTURE WORK

In this paper, we presented a concept for the object-based recognition and assignment of physical properties as density or material based on a 2D image. Our work is motivated by the challenges of distinguishing objects and their properties from each other. The distinction of mass or density enables new interaction possibilities in which the causal relationships of an environment can be linked to the properties of a given object. Our method recognizes specific patterns from 2D images by neural networks in which we estimate the volume by the number of object-recognized triangles. The density is ultimately calculated from the object-specific assignment of a material recognition model and the associated volume.

Despite the promising results of our approach, further work and improvements are needed. An essential aspect relates to the data sets that are used. The accuracy of object-based density recognition goes hand in hand with the quality of the trained AI model. Our current field of application is limited to synthetic test data. Future iterations and evaluations using real-world data sets could help deliver further insights into AI-based density recognition. To achieve serviceable recognition and acceptable results, the appropriate data set needs to be selected for the application area. Therefore, it is necessary to extend the data sets and the training model. The COCO dataset covers numerous categories but neglects existing subcategories. By selecting such data sets, the transferability of the model could be increased.

Our evaluation shows only a limited number of materials assigned to the objects. In reality, the number and composition of materials may be different and more diverse (see Fig. 7). In particular, the interior composition of a recognized object may differ from the recognized surface texture. Consequently, drawing on and combining a wide variety of databases could lead to more precise and serviceable results. These assignments can be linked using an ontological approach. Utilizing ontologies, information and their relationship to each other can be stored in a machine-readable form or made comprehensible. Additional graph databases can visualize the data nodes and their relationship and make them interpretable. In principle, different ontologies can be merged within one ontology. In this context, possible databases on physical properties such as material, geometry and objects could be linked ontologically and applied to the principle of AI-based density recognition.

Supporting these classifications with suitable image segmentation, such as with self-organized maps [17], could further increase the number of distinguishable materials. The partial change of the segmentable areas could be cut out or reduced to densely recognizable areas, which would also reduce quality restrictions and latencies.

Estimating the object size within a scene proves to be a difficult task. This process assumes the same size for all everyday objects in a scene. In the future, it will be necessary to measure the object size and distance of acquired 3D models for meter-level distinctions. The model transfer to a spatial data set would be suitable for this purpose.

The use of image series or video material can also be helpful to support a spatial data set [7]. In this context, the distance and perspective of objects within a scene can be used to determine the speed and possible acceleration of an object. Javadi et al. [11] describe a video-based vehicle speed system for measuring speed based on a measured route. By determining the speed and acceleration of an object, statements can be made

about the forces released in a collision. Video analysis can also be useful for other areas of physical property recognition. The depiction of an object in several individual images with different perspectives allows properties derived previously to be checked and re-evaluated. This includes, for example, the volume or size of the object.

8 ACKNOWLEDGEMENTS

We thank Philip Raschdorf for his support during concept development and data collection. We also thank Thomas Odaker and Elisabeth Mayer, who supported this work with helpful discussions and feedback.

9 REFERENCES

[1] Flickr material database (fmd). MIT (2024-05-06), https://people.csail.mit.edu/celiu/CVPR2010/FMD/index.html

[2] Bell, S., Upchurch, P., Snavely, N., Bala, K.: Material recognition in the wild with the materials in context database. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 3479–3487 (2015). https://doi.org/10.1109/CVPR.2015.7298970

[3] Cai, Z., Fan, Q., Feris, R., Vasconcelos, N.: A unified multi-scale deep convolutional neural network for fast object detection. In: Computer Vision - ECCV. pp. 354–370 (2016). https://doi.org/10.1007/978-3-319-46493-0_22

[4] Daglioglu, M.A.: Object-detection mit you only look once (yolo) : Einführung in die objekterkennung mit yolo sowie die weiterentwicklung in den versionen v2-v4 (2021)

[5] Erdem, K.: Understanding region of interest (roi pooling) (2020), https://erdem.pl/2020/02/understanding-region-of-interest-ro-i-pooling, (visited on 2024-02-20)

[6] Ester, M., Sander, J.: Knowledge discovery in databases. Springer (2000). https://doi.org/10.1007/978-3-642-58331-5

[7] Evain, A., Khemmar, R., Orzalesi, M., Ahmedali, S.: Impact of calibration matrices on 3d monocular object detection: Filtering, dataset combination and integration of synthetic data. International Conferences in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG) (2024)

[8] Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 1440–1448 (2015)

[9] Ippolito, P.P.: Feature extraction techniques (2019), https://towardsdatascience.com/feature-extraction-techniques-d619b56e31be, (visited on 2024-02-20)

[10] Janiesch, C., Zschech, P., Heinrich, K.: Machine learning and deep learning. Electronic Markets **31**(3), 685–695 (2021). https://doi.org/10.1007/s12525-021-00475-2

[11] Javadi, S., Dahl, M., Pettersson, M.I.: Vehicle speed measurement model for video-based systems. Computers & electrical engineering **76**, 238–248 (2019). https://doi.org/10.1016/j.compeleceng.2019.04.001

[12] Jocher, G., Chaurasia, A., Qiu, J.: Ultralytics yolov8 (2023), https://github.com/ultralytics/ultralytics, (visited on 2024-02-20)

[13] Krig, S.: Computer vision metrics: Survey, taxonomy and analysis of computer vision, visual neuroscience, and deep learning. Springer (2016)

[14] Liu, C., Sharan, L., Adelson, E.H., Rosenholtz, R.: Exploring features in a bayesian framework for material recognition. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2010). https://doi.org/10.1109/CVPR.2010.5540207

[15] Mehlig, B.: Machine learning with neural networks: an introduction for scientists and engineers. Cambridge University Press (2021)

[16] Müller, J., Fregin, A., Dietmayer, K.: Disparity sliding window: Object proposals from disparity images. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 5777–5784 (2018). https://doi.org/10.1109/IROS.2018.8593390

[17] Müller, S., Kranzlmüller, D.: Self-organising maps for efficient data reduction and visual optimisation of stereoscopic based disparity maps. In: International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (2022). https://doi.org/10.24132/CSRN.2021.3101.3

[18] Nicodemus, F.E.: Directional reflectance and emissivity of an opaque surface. In: Applied Optics (1965). https://doi.org//10.1364/AO.4.000767

[19] Nielsen, M.A.: Neural networks and deep learning. Determination press, San Francisco, CA, USA (2015)

[20] Redmon, J., Farhadi, A.: Yolo9000: Better, faster, stronger. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017). https://doi.org/10.1109/CVPR.2017.690

[21] Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. Tech. rep. (2018). https://doi.org/10.48550/arXiv.1804.02767

[22] Ren, S., He, K., Girshick, R., Jian, S.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: IEEE transactions on pattern analysis and machine intelligence. pp. 1137–1149 (2017). https://doi.org/10.1109/TPAMI.2016.2577031

[23] Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. nature **323**(6088), 533–536 (1986)

[24] Shalev-Shwartz, S., Ben-David, S.: Understanding machine learning: From theory to algorithms. Cambridge university press (2014)

[25] Shukla, A., Kalnoor, G., Kumar, A., Yuvaraj, N., Manikandan, R., Ramkumar, M.: Improved recognition rate of different material category using convolutional neural networks. Materials Today: Proceedings **81**, 947–950 (2023)

[26] Sun, Y., Gu, Z.: Using computer vision to recognize construction material: A trustworthy dataset perspective. In: Elsevier (2022). https://doi.org/10.1016/j.resconrec.2022.106362

[27] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1–9 (2015). https://doi.org/10.1109/CVPR.2015.7298594

[28] Wu, J., Lim, J.J., Zhang, H., Tenenbaum, J.B., Freeman, W.T.: Physics 101: Learning physical object properties from unlabeled videos. In: Proceedings of the British Machine Vision Conference (2016)

[29] Yadav, N., Yadav, A., Kumar, M.: An introduction to neural network methods for differential equations. Springer (2015). https://doi.org/10.1007/978-94-017-9816-7