One-shot recognition of any material anywhere using contrastive learning with physics-based rendering

Manuel S. Drehwald[3,*,#], Sagi Eppel [1,2,4,*,#], Jolina Li [2,4], Han Hao[2], Alan Aspuru-Guzik[1,2,#]

Abstract

Visual recognition of materials and their states is essential for understanding the world, from determining whether food is cooked, metal is rusted, or a chemical reaction has occurred. However, current image recognition methods are limited to specific classes and properties and can't handle the vast number of material states in the world. To address this, we present MatSim: the first dataset and benchmark for computer vision-based recognition of similarities and transitions between materials and textures, focusing on identifying any material under any conditions using one or a few examples. The dataset contains synthetic and natural images. Synthetic images were rendered using giant collections of textures, objects, and environments generated by computer graphics artists. We use mixtures and gradual transitions between materials to allow the system to learn cases with smooth transitions between states (like gradually cooked food). We also render images with materials inside transparent containers to support beverage and chemistry lab use cases. We use this dataset to train a Siamese net that identifies the same material in different objects, mixtures, and environments. The descriptor generated by this net can be used to identify the states of materials and their subclasses using a single image. We also present the first few-shot material recognition benchmark with natural images from a wide range of fields, including the state of foods and beverages, types of grounds, and many other use cases. We show that a net trained on the MatSim synthetic dataset outperforms state-of-the-art models like Clip on the benchmark and also achieves good results on other unsupervised material classification tasks. Dataset, generation code and trained models have been made available at: https://github.com/ZuseZ4/MatSim-Dataset-Generator-Scripts-And-Neural-net

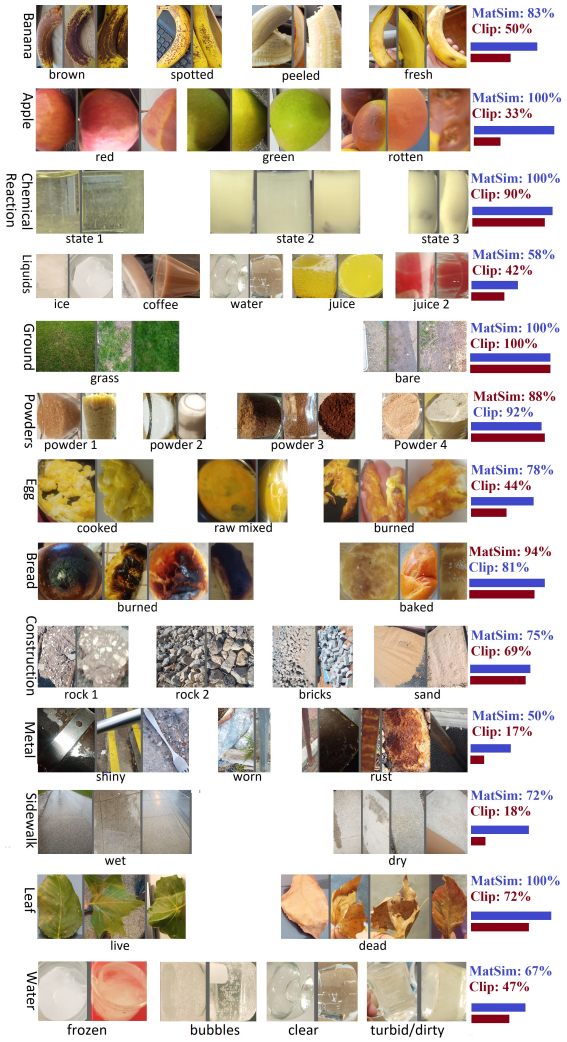

Figure 1. The MatSim benchmark for identifying materials from every aspect of the world using one or a few natural images (few-shot learning). Top-1 results of ConvNeXt trained on the MatSim dataset and pretrained Clip H14, on materials classes unseen during training. Only selected samples are shown.

1. Introduction

The ability to visually identify materials is critical for a wide range of applications, from material science and

[1]Vector institute, [2]University of Toronto,
[3]Karlsruhe Institute of Technology, [4]Innoviz
* Equal Contributions, # Corresponding authors
alan@aspuru.com, manuel@drehwald.info, sagieppel@gmail.com

Figure 2. Selected examples from the MatSim synthetic dataset. The upper line contains random materials on random objects. The bottom row contains random materials inside transparent vessels.

chemical research to cooking, construction, and industry (Figure 1) [31, 38, 22, 39, 14, 15]. Recognition of materials and their states is essential for handling and inspecting materials in the chemistry lab, evaluating whether the ground is wet, determining whether a fruit is ripe, detecting rust on metal surfaces, and distinguishing between different types of rocks or fabrics. There are several major challenges that make this problem difficult for computer vision methods. First, there are almost infinite material states and textures, and each can look very different in different settings. The second challenge is that transitions between material states tend to be gradual and have a continuous intermediate state, which makes it hard to use discrete categories to describe the material (like cooked food). Another challenge is that liquids and materials in environments such as laboratories, hospitals, and kitchens are usually handled inside transparent vessels that distort the view of the materials. Previous studies on image recognition for materials have focused on distinguishing between material classes, such as metal, plastic, and wood, or determining properties like turbidity or glossiness. The limitation of these approaches is that they can only work on the classes and properties they were trained on [13, 8, 16, 29, 36, 14, 7]. To our knowledge, the problem of identifying unseen materials in any environment using one example (one-shot learning) has not been addressed by any benchmark or dataset. In this work, we propose the first general dataset and benchmark for one-shot recognition of any material state in any environment using only one or a few examples.

1.1. The MatSim Dataset

The MatSim dataset includes large-scale collections of synthetic images (Figure 2) for material self-similarity and a diverse natural image benchmark to test the ability of the net to identify material states and subclasses using one or a few examples (Figure 1). This dataset is designed to address the general issue of one-shot material retrieval without restrictions on material types, settings, and environments. The main focus is on distinguishing between states of materials and identifying fine-grained categories such as rotten vs. ripe or coffee vs. cocoa. Additionally, we created a second adversarial benchmark to test the net's ability to recognize materials without association with objects or environments. This benchmark involves covering objects with random materials to create uncorrelated material-object associations.

1.2. Synthetic Dataset and Training

Our hypothesis is that training a Siamese net to identify the same material texture on different objects and environments will allow the net to recognize materials in any setting. While material types are not always matched to a single texture, we assume that a diverse enough training set will force the net to learn a general representation of materials and their properties. The main advantage of this self-similarity approach is that when applied to synthetic data, it can be used to generate an unlimited amount of data with no human effort. In addition, it can be easily expanded to materials mixtures and gradual transitions. The main challenge is the need for a large and highly diverse dataset to prevent the net from overfitting to specific materials or environments.

1.3. Evaluation and Results

The net trained on the MatSim dataset achieves good results in recognizing and matching materials of the same states and subclasses, outperforming state-of-the-art nets like CLIP and nets trained on human-annotated similarity metrics. We also demonstrate that the net performs well in matching images of the same general class in standard classification datasets such as OpenSurface[7] and DMS[32], without using the semantic class labels, suggesting that it learned robust descriptors that generalize beyond its original task.

1.4. Contribution

1) This work introduces the first general dataset and benchmark for low-shot material recognition. The baselines tested in the experimental section demonstrate that the dataset achieves this, allowing the net to identify subclasses, gradual transitions, and materials states, and even allowing the net to generalize to different tasks, such as material classification. 2) Demonstrating that a net trained only on synthetic data and self-similarity, using a single GPU,

can outperform large-scale foundation models like Clip on a general low-shot problem. 3) Show how large-scale CGI assets repositories can be combined to create highly diverse synthetic training data with diversity far exceeding existing materials data sets (hundreds of thousands of different objects and textures) and can be scaled indefinitely using AI textures/objects generating techniques.

2. Related work

2.1. One-Shot and Contrastive Learning

Recent years have seen the emergence of powerful one-shot methods like Clip, which rely on contrastive learning to understand image and text similarities.[26, 27]. The main advantage of these approaches is that, when trained on enough data, they are not limited to a specific set of cases and can work with new examples and classes [26, 17]. These nets work by predicting a descriptor vector for an image and using the distance between different image descriptors as a similarity metric. The limitation of this approach is that it requires an enormous amount of training data, as seen in the Laion Clip H14 model, which was trained on two billion images and their text captions [27]. An alternative, such as SimClr[12], leverages self-similarity between images and their augmented versions, bypassing the need for human-made captions, which offers an unlimited amount of training data. This work takes a similar approach and assumes that, by learning to identify the same material in different physical settings and mixtures, the net will learn a general descriptor for materials and their states.

2.2. Image Retrieval and Materials Similarity

The problem of one-shot material recognition and material matching has been mostly ignored from a computer vision perspective. However, it is closely related to the problem of image retrieval and similarity, which involves finding a similar example from a set. Two such approaches were suggested to identify the visual similarity of simulated materials for the replacement of CGI materials. Schwartz and Nishino[28] trained a net to measure the similarity of materials using 9,000 generated images with uniform materials. The visual similarity between the simulated materials was determined by human annotators. Perroni et al.[23] used transfer learning to determine visual similarity by using the inner layer of a net trained in classification tasks with eight classes of materials. Both studies generate material descriptors to assess similarity. However, both were confined to a specific domain of CGI material swapping and were not intended for material recognition. Consequently, the human-guided net[28] performed poorly in the real-world material recognition tasks we tested, despite this we use it as a baseline due to the lack of alternatives. The other net has not been made available[23].

2.3. Computer Vision for Materials

Computer vision for materials has focused mainly on a few problems: Material segmentation which involves finding the region of the image belonging to specific materials. Inverse rendering, which involves extracting textures maps from object surfaces (for CGI purposes). Predicting the values of specific properties of the material, such as glossiness or turbidity[10, 35, 20, 34, 35, 18, 28]. Neither of these methods is used for one-shot material recognition or can be directly applied to this problem.

2.4. Materials Classification

Materials classification involves assigning a class for the material in the image from a set of categories in the training sets. Datasets for general material classes (wood, metal, glass) include CURet[13], Flicker-Materials FMD[29], KTH-TIPS [16], MINC[8] and large-scale, diverse datasets like OpenSurfaces[8, 7], and Dense Material Segmentation (DMS)[32]. Other datasets focus on more specific properties, such as material phases (liquids, solids, powders, foam)[14], and more specific classes, such as construction materials [9], soil type [30], crystallization, turbidity[25], etc.[21, 31, 38, 22, 15]. The limitation of nets trained on these datasets is that they are limited to specific classes in the dataset and cannot work on classes not used during training.

3. The MatSim Synthetic Dataset Generation

The goal of the MatSim dataset is to train a net capable of recognizing any visually distinguished material in any setting. The visual appearance of a material depends not only on its physical properties but also on the object's shape, environment, and illumination. If during training, any of these properties are restricted in some way (for example, only indoor scenes are used) or correlated with other properties (for example, wood materials appear only in trees), it is unlikely that the net achieves a true generalization for material recognition [17, 5].

3.1. Generation Procedure

In order to achieve the above goals, the dataset must be extremely diverse in terms of objects, environments, and materials. To achieve this, we utilized large-scale CGI repositories used for animation and computer games. We used thousands of highly diverse physics-based rendering materials (SVBRDF / PBR) from the AmbientCG, CGBookCase, and FreePBR repositories [1, 37, 2] to simulate realistic materials[24]. We overlay the textures materials on 3D objects taken from the ShapeNet dataset with tens of thousands of different objects [11] and hundreds of categories; these objects are then placed in random scenes with natural illumination and backgrounds taken from the Poly-

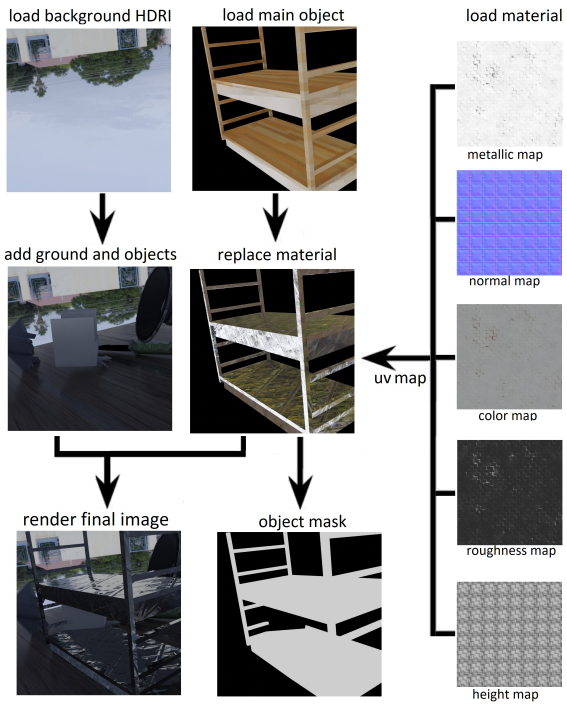

Figure 3. Dataset creation: 1) CGI Materials have been randomly created or downloaded from large-scale artist repositories (AmbientCG[1], CGBookCases[37], FreePBR [2]). 2) The material is UV mapped on the surface of a random object loaded from the ShapeNet dataset[11].3) Random background and illumination are loaded from the HDRI Haven repository[3]. 5) Ground plane and background objects are added. 6) The scene is rendered.

Figure 4. Dataset structure. The dataset is composed of sets. Each set involves two materials and six scenes with a gradual transition between the two materials. Each column in the image corresponds to a different mixing ratio (R) of the two materials. The ratio of the mixture of the two materials is given in the top column. All images in the same column involve the same mixture ratio (R) on different objects and in different environments. Each of the six scenes involves one main object. The material on this object gradually transitions from one material to another. The mask of the object is given in the right column. For scenes 1–2, the background remains exactly the same for all images in the scene. For scenes 3–4, the background HDRI is randomly rotated between images, leading to small changes in illumination. For scenes 5–6, the background HDRI is completely replaced between images, leading to large changes in illumination.

hHaven repository for HDRI images[3]. The HDRI image is wrapped around the scene and provides a realistic 360-degree background and illumination (Figure 3). Combining these repositories allows us to generate a large-scale, highly diverse dataset (Figure 3). The large number of materials forces the net to generalize rather than identify only specific classes. The fact that every material can be used on any object in any environment means that the net has to identify the material everywhere and prevents the net from associating the material with specific objects or environments. Gradual transformations between materials in the dataset allow the net to detect gradual transitions between materials. Additionally, rendering some of the materials inside transparent containers allows the net to learn to recognize materials stored inside glass vessels. Finally, in some scenes, light sources were scattered in random positions to simulate near-field illumination.

3.2. General Dataset Structure

The dataset is divided into sets; each set contains two random materials and six scenes that involve a gradual transition between these two materials (Figure 4). The objects, backgrounds, and environment are randomly selected for each scene separately (Figure 3). Each scene involves one static main object and a static camera, both positioned randomly. The object material is gradually changed between images in the scene (Figure 4). For each scene, we render five images with different mixtures ratios (R) of the two materials (0%, 25%, 50%, 75%, 100%). 0% means that the object is made only of material A, while 100% means that it is made completely of material B (Section 3.3). We change the illumination of the environment and objects between scenes, which provides a wide range of variations in each material's appearance. Since all scenes in a set are composed of the same two materials, it is possible to compare the appearance of the materials between two scenes with different objects, backgrounds, and light. The gradual transition allows the net to learn to distinguish between highly similar materials and mixtures. Since a scene can contain many background objects, for each scene, we provide the mask (region) of the object on which the material is used (Figure 4, right). Around 30k sets, with about a mil-

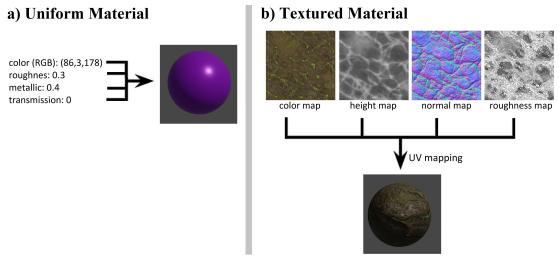

Figure 5. A material's visual appearance is controlled by several main properties (color, transparency, etc.). For uniform materials (BSDF), each property has a single value across the surface. Textured materials are represented by texture maps for each property. These maps are wrapped around the object (UV mapping) and provide properties for each point on the object's surface.

lion images, were rendered. For more details, see Appendix 9.

3.3. Materials Representation, Mixtures, and Gradual Transformations

The appearance of materials is mostly controlled by their surface scattering properties. These properties are often referred to as bidirectional reflectance (BRDF) and the more general bidirectional scattering function[6, 4] (BSDF). Each surface point has a set of properties (roughness, transmittance, color, etc.) that determine the surface appearance. If the material is uniform, it could be represented as a set of values for each property across the entire surface (Figure 5a). Creating a random uniform material is done by setting a random value for each property. Mixing two such materials can be achieved by using a weighted average of the values of each property of each material: $P_{\text{mix}} = R \cdot P_a + (1 - R) \cdot P_b$ where P is the value of a property in materials a,b and R the mix ratio. Most materials in the world are not uniform and have unique textures, which means different properties for each point on the surface. Such materials can be represented as spatially variable BRDF or SVBRDF often called PBRs ([24]). This means that instead of a single value for each property, we have a 2D texture map that represents the spatial distribution of this property on the surface. This 2D map is then wrapped around the object to give each surface point its property (Figure 5a). The mixing of two textured materials is achieved by a pixel-wise weighted average of two texture maps into a new texture map. In other words, each point in the texture map of a given property is the average of the corresponding pixels in the two materials that are mixed: $P_{\text{mix}}(u, v) = R \cdot P_a(u, v) + (1 - R) \cdot P_b(u, v)$. Where $P(u, v)$ is the property value in the surface position u,v. The gradual transition between materials A and B is again achieved by setting different ratios of mixtures (R). To increase the variability, texture maps of different materials can be rotated and rescaled relative to the other material before

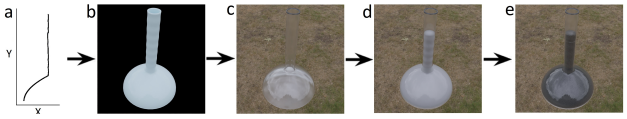

Figure 6. Procedurally generating material inside transparent containers. a) A random 2D curve is generated by combining random polynomial and trigonometric functions. b) The curve is used as a profile for the symmetric 3D object. c,d) The object is assigned random transparent materials and content object is generated inside. e) The content object is assigned a material.

mixing. Unlike uniform BSDF textures, it is not possible to create textures by assigning random values. Therefore, we obtained a large number of textured materials from large-scale artists' repositories. To further increase this set, we mix two or more materials as described above.

3.4. Materials in Transparent Vessels

Liquids and other materials in kitchens, hospitals, and labs are usually handled inside transparent containers (glassware, flasks, tubes, etc.). To support these applications, we generated scenes in which we put the material inside a transparent container. The vessel object was procedurally generated. The curvature of the transparent vessel was generated by creating a random 2D curve (by randomly combining linear, polynomial, and trigonometric functions; Figure 6). This curve was used as the profile of a symmetric vessel by creating a cylindrical (or other symmetrical) shape with the curve as the vertical profile (Figure 6). In some cases, the shape was also randomly stretched to create more variability. The vessel object was assigned a random transparent material and a random thickness. The content of the vessel was either a random object loaded from ShapeNet or a mesh that filled the bottom part of the vessel (similar to a static liquid or powder). As before, the content was assigned random material. Otherwise, the creation of the data set was the same as in Section 3.2.

4. Benchmark Creation

Perhaps the most important part in dealing with the task of general material recognition is to define a proper benchmark that covers the main aspects of this challenge. We define three main capabilities that we want to evaluate: 1) The ability of the net to visually identify unfamiliar materials in new environments using one or a few examples with no restrictions on the material or environment setting. 2) The ability to identify transitions between material states (e.g., wet/dry, rotten/fresh) and fine-grained subcategories of materials (e.g., types of rocks). 3) The ability to identify materials regardless of the object on which they appear (for example, a cow made of wood)[5, 17]. We are not aware of a benchmark that covers any of these three tasks. Therefore, we created two benchmarks by taking pictures of materials

in a variety of states and environments.

4.1. Set 1: Material Transition States and Subclasses

The first test set is designed to test the hypothesis that the net is capable of recognizing the similarity between images of the same material, even when they are presented in transparent containers and different surroundings, while distinguishing between subcategories or different states of the same material. The focus of this test set is real-world objects and scenes in a wide range of settings. The only intervention was that, when possible, samples of the same materials were moved to a new scene to change environments. This set contains a large number of examples of materials that change in a continuous manner (wet/dry, cooked/uncooked). And tests the net ability to deal with continuous transitions by assigning material to its closest state (Figure 1); for example, we collected images of eggs in raw, cooked and burned states and sidewalks in wet and dry states (Figure 1). For each state, we collected at least two different images. We divided this test set into superclasses (e.g., eggs) and divided each superclass into subclasses (e.g., cooked/raw/burned; Figure 1). We try to make different examples of the same material subclass appear in different samples and environments as much as possible, making the recognition of the material based on the type of environment or object less likely. Additionally, for each image, we created a mask of the material region in the image. Note, however, that in this set there will always be some correlation between materials and objects, i.e., liquids appear in glassware and rotten textures appear on fruits. This set contained 416 images divided into 116 material types.

4.2. Set 2: Uncorrelated Materials and Objects

The second test set is designed to test the hypothesis that the net can recognize the material regardless of the object on which it appears. Materials in the real world are strongly correlated to objects (trees made of wood). Creating this set meant that we needed to generate our own objects. To achieve this, we collected various objects and materials, including ribbons, sheets, and granular materials. These were adhered to the objects' surfaces, either by wrapping or scattering, using glue. When creating this test set, we made sure that each material did not appear on the same object or in the same environment in more than one image, forcing the net to use only the material for recognition. This set contains 86 images in 16 material types.

4.3. Materials Classification Using a Descriptor

The MatSim dataset relies on visual similarity. It is not intended and is less suited for handling broad semantic material classes. General classes, like wood or plastic, may include a wide range of textures with little or no visual similarity between them. However, recent studies have shown that nets trained on self-similarity can often generalize effectively to broader semantic classes when provided with a large set of images [33]. Although not the primary focus of our work, we tested the MatSim-trained net and CLIP[26, 27] on major material classification datasets, to see how well they generalize beyond their core task. The evaluation process followed the same method used for other MatSim benchmarks (Section 5), where each image in the data set was matched to all other images. If the best-matched image belonged to the same class, it was considered correctly classified. The accuracy of each class was calculated as the average of all images in that class. The overall accuracy was determined as the average accuracy for all classes. We examine the main datasets for materials classification: OpenSurface[8, 7], Flicker Materials (FMD)[29], and Dense Material Segmentation(DMS)[32]. We use the region/segment of the material as the net input mask (Figure 7). For CLIP, we also tested the net ability to classify materials by matching the image with the text label (standard semantic classification[26, 27]).

5. Evaluation Methods

For the evaluation of the net, we consider the standard Top-1 accuracy metric: Given an image of a material, we test the ability of the net to identify another image of the same material from a group of images. An example is correct if the highest similarity calculated by the net is between the given image and another image of the same material subclass. We calculate two accuracy metrics. The first evaluation method assesses the ability to identify another instance of the same material among all samples from the same superclass (Figure 1). For example, the task could be to identify an image with spotted bananas from a set of images of bananas in various states (Figure 1, Table 1 (subclasses)). In the second evaluation method, we still expect the net to find another example of the same material subclass. However, we now present it with all the other images in the test set, including those of different superclasses. For example, the image of a spotted banana is compared to all images, including those of rust, rocks, cheese, etc. For the second test set and other material classification datasets, we used only the second evaluation approach (no superclasses). The results given in Table 1, Figure 1, are the average precision per class (all classes given equal weights).

6. Training and Net Architecture

6.1. Net Structure

The net architecture follows that of Clip/SimClr[12, 26], with a ConvNeXt[19] encoder that receives an image and a mask of the material region stack together as a 4-layer input (R,G,B,Mask), and outputs a descriptor vector of length

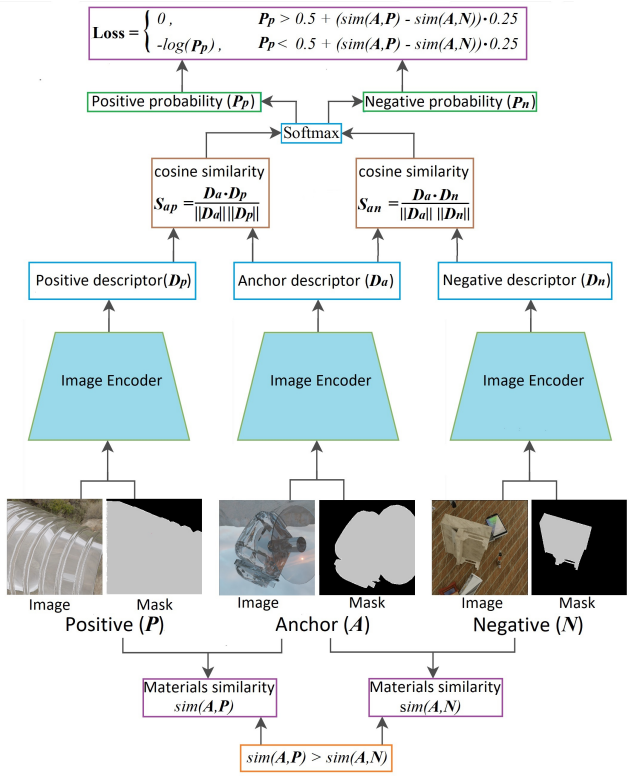

Figure 7. Training and loss function. The loss function is based on cosine similarity with cross-entropy loss. For every three images in the batch, one image is defined as an anchor (A). A second image, specifically the image that has material more similar to the anchor, is defined as positive (P), and the remaining image is defined as negative (N). All images and their material masks are passed through the neural net to produce descriptors (D_a, D_n, D_p). Cosine similarity is calculated between the descriptor of the anchor and the positive and negative examples. These cosine similarities are the input to the softmax function, which returns the probability of a match between the anchor and the positive image (P_p). If this probability is below the threshold defined as $0.5 + (sim(A,P) - sim(A,N)) \cdot 0.25$, we calculate the cross-entropy loss ($Loss = -\log(P_p)$); otherwise, the loss is set to zero. $sim(A,P) = 1 - |R(A) - R(P)|$ is the similarity between the anchor (A) material and the positive material P, $R(A)$ is the mixture ratio in material A (Section 3.3).

512 (with L2 normalization). The cosine similarity between two predicted descriptors is used to assess the similarity between two materials (Figure 7).

6.2. General Training

Training was carried out as shown in Figure 7. For each batch, we randomly selected a single set (Figure 4) and sampled 12 random images from this set. The loss for every three images in the batch was calculated separately: One of the three images was chosen as anchor, the image that is more similar to the anchor in terms of the mixture ratio (R, Figure 4,Section 3.3) is used as positive and the remaining image as negative. We pass each image through the net to predict its descriptor. We calculate the cosine similarity between the descriptors of the anchor and both negative and positive. These similarities are passed to a Softmax to calculate the positive and negative match probabilities (how well the anchor material matches the positive and negative material). This is then used as standard classification probabilities and trained using cross-entropy loss.

6.3. Loss Function and Materials Similarity

The level of similarity between materials is continuous and can have any value between zero and one. Therefore, the anchor material is not necessarily 100% identical to the positive material, but can, for example, have 50% similarity to the positive and 25% similarity to the negative. We define the similarity of two materials in a set as the difference between their mixture ratios (R, Figure 4, Section 3.3):

$$sim(i_1, i_2) = 1 - |R(i_1) - R(i_2)|$$

Where $R(i_1)$ is the mixture ratio in the image i_1. By definition, the similarity of the anchor (A) to the positive (P) material is larger than to the negative (N) ($sim(A,P) > sim(A,N)$). Therefore, we expect that the predicted probability (P_p, Figure 7) for a match between the anchor (A) and the positive (P) will be higher than 50%. However, we do not expect it to be 100% if the similarity of the anchor and positive ($sim(A,P)$) is near that of the anchor and negative ($sim(A,N)$). Therefore, we set a semi-hard loss with the following condition: If $P_p > 0.5 + (sim(A,P) - sim(A,N)) \cdot 0.25$, the loss is set to zero; otherwise, we calculate the standard cross-entropy loss ($Loss = -log(P_p)$). More details can be found in Appendix 10.

7. Testing Clip Models

The Clip neural network has achieved state-of-the-art (SOTA) results on a large number of image recognition benchmarks and is considered one of the top general one-shot nets to date [27, 26]. Similarly to our model, Clip predicts image descriptors as a vector, which can then be used to find similarities between the image and other images or texts. Unlike our model, Clip H14 was trained by comparing 2 billion real photos with their corresponding text captions. There are two limitations to using Clip for this task. The first is that the Clip descriptor is not focused on materials and might contain information regarding the object and environment. Second, Clip receives just an image without an attention mask. Hence, it is harder to point to a specific region of the image where the target material is. To solve this issue, we tried a few methods: The first involves cropping the region of the image around the target material and using it as input. The second involves masking the image region outside the material (by setting it to black), and

Method	Set1 Subclass	Set1 All	Set2
Random	0.30	0.006	0.07
MatSim	0.71	0.56	0.73
MatSim+C	0.77	0.56	0.85
MatSim+M	**0.78**	0.56	**0.91**
MatSim+C+M	0.72	**0.61**	0.85
Open CLIP H14	0.55	0.44	0.47
Open CLIP H14+C	0.67	0.52	0.77
Open CLIP H14+M	0.59	0.40	0.53
Open CLIP H14+C+M	0.66	0.52	0.67
CLIP B32	0.51	0.32	0.44
CLIP B32+C	0.56	0.38	0.49
CLIP B32+M	0.56	0.28	0.37
CLIP B32+C+M	0.56	0.35	0.56
Human Similarity	0.41	0.13	0.22
Human Similarity+C	0.60	0.20	0.41
Human Similarity+M	0.55	0.15	0.27
Human Similarity+C+M	0.55	0.16	0.42
MatSim+M No Augment	0.66	0.45	0.61
MatSim+M No vessels	0.72	0.52	0.61
MatSim+M No mixtures	0.74	0.53	0.72

Table 1. Results. $+C$ indicates cropping, $+M$ indicates masking. Random stands for random matching. Human Similarity refers to the net trained on human-annotated material similarity metrics[18]. No vessels refer to a net trained only on the objects part of the MatSim dataset. No mixtures refer to a net trained on MatSim without materials mixtures.

the third involves combining masking and cropping. All of these approaches gave a significant improvement compared to using the image as is. Cropping without masking gave the best results. We tested all the pretrained Clip versions available and found that ViT-B32 gave the best results for Open A.I models (Table 1), but the Laion Clip H/14 significantly outperforms all other Clip Models[27].

8. Results

As seen in Figure 1 and Table 1, the MatSim trained net performed well in almost all types of materials and environments. The MatSim-trained net significantly outperforms the best Clip model on all test sets (Table 1). This suggests that despite the fact that the net was trained only on simulated data and visual self-similarity, it learned to generalize to unfamiliar real-world material states. Both Clip and MatSim significantly outperformed net trained on human-annotated material similarity metrics[18]. All approaches significantly outperformed random matching (Table 1). Both MatSim-trained net and Clip performed well on Set 2 despite this set having no correlation between materials and objects. This supports the hypothesis that recognition is done only based on material features and not correlated properties. The MatSim trained net gives good results on cases with near-field light sources like floors (100%), but the number of such cases in the benchmark is small. Focusing the net on the material region of the image by cropping the material region and using it as the input image seems to improve the accuracy of all the nets tested (Table 1). Masking the background by blacking out the background pixels seems to improve the results for the MatSim trained net, but proves to be less effective than cropping for Clip.

Method	OpenSurface	DMS	FMD
Random	0.03	0.019	0.09
MatSim+M	**0.33**	**0.29**	0.66
Open CLIP H14+C	0.32	0.14	**0.82**
Human Similarity+C	0.07	0.04	0.20
Open CLIP H14+C Semantic	0.30	0.19	0.72
Semantic Random	0.03	0.018	0.1

Table 2. Results for Material Classification on Standard Benchmarks(Section 4.3). Random stands for random matching. Semantic stand for matching the image to the text labels. $+C$ indicates cropping, $+M$ indicates masking. The attention mode for each net (masking/cropping) is the one that gives the best results.

8.1. Results for General Material Classification

The MatSim visual similarity approach is not intended for the more general semantic classes, which are associated more by names and less by appearance. However, as can be seen from Table 2, both CLIP and MatSim trained net gave relatively good results for matching images of the same general classes (plastic, glass, wood...). For the DMS[32] and OpenSurface[8, 7] datasets, the MatSim-trained net outperformed CLIP (Table 2). This could be attributed to the larger number of images and classes, making it more likely for similar textures to occur in each class, while the more diverse class set also makes semantic-guided classification harder. CLIP performed better than the MatSim trained net on the FMD benchmark[29]. This can be explained by the fact that this dataset contains a small set of general classes and a relatively small number of images (100) per class, where each class can contain a large number of different textures, making semantic knowledge of the material class a major advantage, while limiting the effectivity of relying only on textures' visual similarity. The net trained using human-assigned materials similarity[18] performed by far the worst on all datasets but still far above random. We also tested the ability of the CLIP text/image embedding model[26, 27] to classify material images by matching them with their semantic text labels. Clip H14 performs far better than random in this task and near the image-to-image-based matching accuracy (Table 2).

8.2. Training Components and Their Effects

Augmentation, Training for materials inside transparent vessels, and training with a gradual transition between materials are all vital for the performance of the MatSim trained net (Table 1). Eliminating any of these resulted in a considerable decrease in performance, but the net still performed better than CLIP even without these aspects (Table 1).

9. Conclusion

This work introduces a new approach, dataset, and benchmark for general one-shot material recognition, not limited to specific types of material or settings. It tackles several challenges related to material recognition, including material states, subclasses, mixtures, gradual state transition, and recognition of materials inside transparent vessels. One-shot recognition of materials has been mostly overlooked in image recognition research, but it is critical for understanding different aspects of the world and has numerous applications, ranging from material science and chemistry to cooking and agriculture. Our findings demonstrate that a net trained on self-similarity using diverse synthetic data can recognize almost every material state using just one or a few examples, regardless of the environment, outperforming large-scale models trained on human-generated semantic captions (CLIP) and similarity metrics based on human perception. We hope that the data set and the benchmark will pave the way for testing and developing new techniques in this emerging field.

References

[1] Ambientcg, free pbr textures repositories. `https://ambientcg.com/`.

[2] freepbr, free pbr textures repositories. `https://freepbr.com/`.

[3] Polyhaven free hdri repository. `https://polyhaven.com`.

[4] Clara Asmail. Bidirectional scattering distribution function (bsdf): a systematized bibliography. *Journal of research of the National Institute of Standards and Technology*, 96(2):215, 1991.

[5] Laura Alexandra Daza Barragan, Jordi Pont-Tuset, and Pablo Arbelaez. Adversarially robust panoptic segmentation (arpas) benchmark. 2022.

[6] Frederick O Bartell, Eustace L Dereniak, and William L Wolfe. The theory and measurement of bidirectional reflectance distribution function (brdf) and bidirectional transmittance distribution function (btdf). In *Radiation scattering in optical systems*, volume 257, pages 154–160. SPIE, 1981.

[7] Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. Opensurfaces: A richly annotated catalog of surface appearance. *ACM Transactions on graphics (TOG)*, 32(4):1–17, 2013.

[8] Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. Material recognition in the wild with the materials in context database. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3479–3487, 2015.

[9] Ioannis K Brilakis, Lucio Soibelman, and Yoshihisa Shinagawa. Construction site image retrieval based on material cluster recognition. *Advanced Engineering Informatics*, 20(4):443–452, 2006.

[10] Alice C Chadwick and RW Kentridge. The perception of gloss: A review. *Vision research*, 109:221–235, 2015.

[11] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

[12] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[13] Kristin J Dana, Bram Van Ginneken, Shree K Nayar, and Jan J Koenderink. Reflectance and texture of real-world surfaces. *ACM Transactions On Graphics (TOG)*, 18(1):1–34, 1999.

[14] Sagi Eppel, Haoping Xu, Mor Bismuth, and Alan Aspuru-Guzik. Computer vision for recognition of materials and vessels in chemistry lab settings and the vector-labpics data set. *ACS central science*, 6(10):1743–1752, 2020.

[15] Leon Eversberg and Jens Lambrecht. Generating images with physics-based rendering for an industrial object detection task: Realism versus domain randomization. *Sensors*, 21(23):7901, 2021.

[16] Eric Hayman, Barbara Caputo, Mario Fritz, and Jan-Olof Eklundh. On the significance of real-world conditions for material classification. In *European conference on computer vision*, pages 253–266. Springer, 2004.

[17] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8349, 2021.

[18] Manuel Lagunas, Sandra Malpica, Ana Serrano, Elena Garces, Diego Gutierrez, and Belen Masia. A similarity measure for material appearance. *arXiv preprint arXiv:1905.01562*, 2019.

[19] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022.

[20] Abhimitra Meka, Maxim Maximov, Michael Zollhoefer, Avishek Chatterjee, Hans-Peter Seidel, Christian Richardt, and Christian Theobalt. Lime: Live intrinsic material estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6315–6324, 2018.

[21] Perseverança Mungofa, Arnold Schumann, and Laura Waldo. Chemical crystal identification with deep learning machine vision. *BMC Research Notes*, 11(1):1–6, 2018.

[22] Diego Inácio Patrício and Rafael Rieder. Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. *Computers and electronics in agriculture*, 153:69–81, 2018.

[23] Maxine Perroni-Scharf, Kalyan Sunkavalli, Jonathan Eisenmann, and Yannick Hold-Geoffroy. Material swapping for 3d scenes using a learnt material similarity measure. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2034–2043, 2022.

[24] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically based rendering: From theory to implementation*. Morgan Kaufmann, 2016.

[25] Gabriella Pizzuto, Jacopo De Berardinis, Louis Longley, Hatem Fakhruldeen, and Andrew I Cooper. Solis: Autonomous solubility screening using deep neural networks. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE, 2022.

[26] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[27] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *arXiv preprint arXiv:2210.08402*, 2022.

[28] Gabriel Schwartz and Ko Nishino. Recognizing material properties from images. *IEEE transactions on pattern analysis and machine intelligence*, 42(8):1981–1995, 2019.

[29] Lavanya Sharan, Ruth Rosenholtz, and Edward Adelson. Material perception: What can you see in a brief glance? *Journal of Vision*, 9(8):784–784, 2009.

[30] Pallavi Srivastava, Aasheesh Shukla, and Atul Bansal. A comprehensive review on soil classification using deep learning and computer vision techniques. *Multimedia Tools and Applications*, 80:14887–14914, 2021.

[31] Ying Sun and Zhaolin Gu. Using computer vision to recognize construction material: A trustworthy dataset perspective. *Resources, Conservation and Recycling*, 183:106362, 2022.

[32] Paul Upchurch and Ransen Niu. A dense material segmentation dataset for indoor and outdoor scene parsing. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VIII*, pages 450–466. Springer, 2022.

[33] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X*, pages 268–285. Springer, 2020.

[34] Raquel Vidaurre, Dan Casas, Elena Garces, and Jorge Lopez-Moreno. Brdf estimation of complex materials with nested learning. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1347–1356. IEEE, 2019.

[35] Sebastian Weiss, Robert Maier, Daniel Cremers, Rudiger Westermann, and Nils Thuerey. Correspondence-free material reconstruction using sparse surface constraints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4686–4695, 2020.

[36] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European conference on computer vision (ECCV)*, pages 418–434, 2018.

[37] Dorian Zgraggen. Cgbookcase free pbr textures library. `https://www.cgbookcase.com/`.

[38] Song Zhang, Yumiao Chen, Zhongliang Yang, and Hugh Gong. Computer vision based two-stage waste recognition-retrieval algorithm for waste classification. *Resources, Conservation and Recycling*, 169:105543, 2021.

[39] Junwei Zheng, Jiaming Zhang, Kailun Yang, Kunyu Peng, and Rainer Stiefelhagen. Materobot: Material recognition in wearable robotics for people with visual impairments. *arXiv preprint arXiv:2302.14595*, 2023.