Towards Synthetic AI Training Data for Image Classification in Intralogistic Settings

Daniel Schoepflin, Karthik Iyer, Martin Gomse and Thorsten Schüppstuhl

Abstract

Obtaining annotated data for proper training of AI image classifiers remains a challenge for successful deployment in industrial settings. As a promising alternative to handcrafted annotations, synthetic training data generation has grown in popularity. However, in most cases the pipelines used to generate this data are not of universal nature and have to be redesigned for different domain applications. This requires a detailed formulation of the domain through a semantic scene grammar. We aim to present such a grammar that is based on domain knowledge for the production-supplying transport of components in intralogistic settings. We present a use-case analysis for the domain of production supplying logistics and derive a scene grammar, which can be used to formulate similar problem statements in the domain for the purpose of data generation. We demonstrate the use of this grammar to feed a scene generation pipeline and obtain training data for an AI based image classifier.

Keywords

Synthetic data • Training data generation • Image classification • Intralogistic • Production supplying logistic

D. Schoepflin (✉) · K. Iyer · M. Gomse · T. Schüppstuhl
Institute for Aircraft Production Technology, Hamburg University of Technology, Denickestraße 17, 21073 Hamburg, Germany
E-mail: Daniel.Schoepflin@tuhh.de
URL: http://www.tuhh.de/ifpt/

T. Schüppstuhl et al. (eds.), *Annals of Scientific Society for Assembly, Handling and Industrial Robotics 2021*,
https://doi.org/10.1007/978-3-030-74032-0_27

1 Introduction

The intralogistic transport of components on the plant-site of an aircraft manufacturer is comprised of multiple handling and repackaging processes [1]. Such components are often transported on material delivery units, and due to the manual handling, are subjected to error. This is mostly reflected in delivery units being loaded with the wrong components. For this reason identifier tags may be used to verify the loading. However, due to process and manufacturing requirements, tagging the components may be prohibited and a visual object identification system is needed. We consider the usage in a system setting as seen in Fig. 1, where components are placed in boxes in a load carrier and a camera achieves a top-view on those components. Evaluating this top-view raises the need for proper training of an AI image classifier. For the high variety of components in aircraft manufacturing, obtaining and labeling images manually is a tedious and costly process. Thus, synthetic training data is considered a viable alternative.

In recent years, different generation pipelines have been introduced with different fields of application [2–4]. As stated by [5], creating such virtual worlds is in need of domain experts and the formulation of a representative scene grammar. To the authors knowledge, this has not been done sufficiently for the intralogistic transport of components. We therefore contribute a use-case analysis for this domain and define a parametrization that is usable to derive a tool independent scene composition grammar that is transferable to intralogistic use-cases. We implement this in a simulation and rendering pipeline to obtain training data for an AI image classifier.

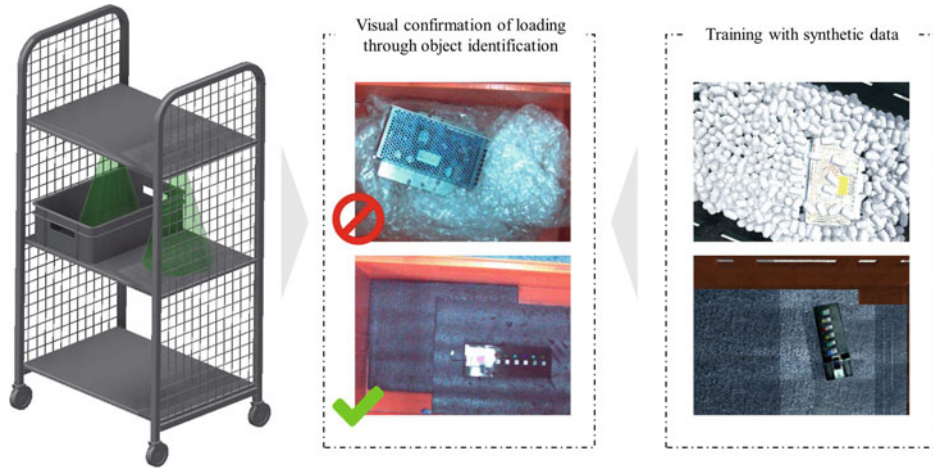


Fig. 1 Use-Case of intralogistic component transport with delivery units. Loading is confirmed with a top view visual object identification system, which is trained with synthetic data

2 Related Work

Various approaches for creation of synthetic driving scenes have lead to multiple synthetic data-sets of different applications like driving scenes [6], household objects [4, 7] to robotic picking [3, 8]. Sets like [8, 9] enable the community to research on industrially relevant box-picking tasks or object identification tasks. A synthetic data generation approach was undertaken by [10] to enable handling of gears and similar components in industrial settings. Due to the success in these domains, it can be inferred that training an object identifier in the highly variational intralogistic scenery with synthetic data is a viable approach. Common object data-sets [2, 4, 11] can be used by multiple users to enable AI services relating to universally common environments e.g. household objects. However, this universal applicability is not necessarily the case for industrial applications or data-sets, as they address non-common objects in specialised environments [12]. Thus, in order to enable broad applicability of synthetic training data approaches, we strive to contribute towards adaptability of generation pipelines with regards to user needs.

In most cases, pipelines place 3D models of the objects of interest in a scene and render this scene to obtain the training image. In general this creation and composition of a scene is varied for each rendering process e.g. changing position and orientation of objects within a pre-defined parameter space. The NVIDIA Deep Learning Data Synthesizer[1] provides a data creation tool based on the Unreal Engine 4, with which the household object datasets SIDOD and FAT were created. Such a tool can be used to create synthetic training data for different domains, when used with a modeled environment or scene creation grammar for that domain. Similar tools can be found for the Unity Game Engine[2] or the Open Source Tool Blender.[3] Although these tools provide easy use for modelled environments, they do not automatically provide semantics for creation of new 3D scenes. This semantic formulation of a scene is mostly provided by the user.

Approaches like [13] use a fully automated and randomized composition of front- and background, whereas approaches like [9] utilize context true scene compositions. In both cases a set of rules is created, on which the implementation of a composition algorithm is based. These accumulated rules and parametrizations are referred to as grammar [5] or model [12] and are vital for the creation of data generation pipelines. We provide such a generic grammar for the use-case of intralogistic transportation of material.

3 Process Analysis and Problem Statement

We first describe the intralogistic transport domain for production supplying logistic with delivery units. To further analyse this, a categorization with respect to the complexity of

[1] https://github.com/NVIDIA/Dataset_Synthesizer, September 2020.
[2] https://github.com/Unity-Technologies/SynthDet, September 2020.
[3] https://github.com/921kiyo/3d-dl, September 2020.

scenes is developed. This then leads to a generalized grammar for this domain. To validate this grammar, a parametrization for later implementation of a specific use-case is formulated.

3.1 Use-Case Description

As the generated data shall enable the training of an AI image classifier, the scope of this specific application is defined. With the considered use as shown in Fig. 1, the main application that a visual based AI image classifier may enable is the validation of commissioning. Through identification of the loaded component and comparison with an expected loading, the validity of loading can be concluded. This may be combined with a counting of the objects. However these tasks can not be realised for every component type. The components have to be visually detectable and thus, can not be fully wrapped in packaging or appear in great numbers with mutual concealment. Thus, the likely number of objects to identify in one scene is in the single to low double digit range. The components are often placed in boxes, with little position and pose constraints or on shadowboards with pre-defined positions. Further we assume that the identification task is applied to components that may not carry RFID or similar identification markers. This is mostly the case for assembly ready components and lesser for semi-finished products. Figure 2 visualizes these constraints of the possible scenarios which we aim to synthesize.

As seen in Fig. 1, the considered setting of the cameras on a delivery unit itself provides a top-view of the loaded components. However, in other cases an askew view may be provided, e.g. with a camera above a robotic effector. Thus it is neccessary, that the derived scene grammar will take such cases in consideration.

Fig. 2 Definition of the scope for the approach considered in this use-case. Objects appear in the visually countable numbers and are visually detectable

3.2 Categorization of Transportation Settings

As the sceneries to which the AI is applied may differ significantly, a categorization is undertaken. This enables a more detailed analysis of the domain and then leads to the description of the domain in a more generalized grammar formulation. These categories are defined regarding the complexity of parameter variations occuring in these settings:

1. **Simple settings**: objects and components are placed on a uniform background such as anti-slip mats. Besides material changes leading to differences in texture and color of the background, parameter variations occur with the placing of the object on such a mat. Objects can occur in different translatory positions as well as rotational placements. However, many components are limited in their contact points to the mat, to which they are forced by their axes of inertia and gravity. This leads to discrete number of stable resting states, with only one axis as rotational degree of freedom.
2. **Intermediate settings**: many components, in particular small components, are transported in boxes or cartonages. Those are also subject to the same placement restrictions as the simple cases above. Causing a more complex scenery is the variant lighting condition as well as the shadowing caused by the boxes.
3. **Complex settings**: some shock sensitive components may be transported in boxes filled with packaging flips or in bubble wrap. Some flat components are transported by placing them between struts of a support structure. Additional straps may be used to secure the components. Besides creating a complex scenery by adding a complex setting of other objects, they also may allow the components to be placed in different positions than the previous two settings. Additionally, lighting situations are more complex, due to local shadowing.

3.3 Formulation of Scene Grammar

It is now necessary, to define a scene composition grammar with respect to the parameters of three above presented categories. We first focus on the object composition and briefly introduce further variations like background, view and lighting.

3D Composition Semantic

Arrangement of real components on a delivery unit follows a set of semantic rules, that are mostly intuitively met by the persons handling the commissioning. In order to later simulate a loading scenario, relations between the objects to place have to be defined, which together forms a semantic 3D composition grammar. This is displayed in Fig. 3: We start with the type of unit in use. This defines the geometrical restrictions in which the boxes and components are placed. Further, this defines the location and type of the camera and further

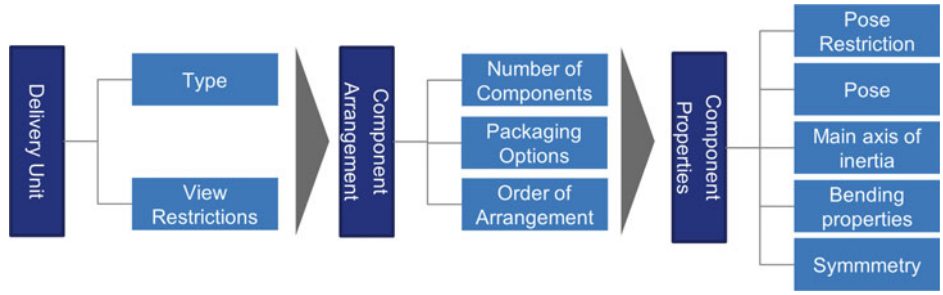


Fig. 3 Sequence of the semantic scene grammar parameters with respect to the object composition relation

view restrictions. Afterwards, the principal component arrangement has to be defined. The number of components to be modelled is defined and their relative arrangement set. This is done with the components properties as well as possible packaging options. Those packaging options refer to the arrangement of components in a box, packaging material like flips and possible distractors like transfer papers. Afterwards for each component the individual pose spectrum has to be defined. As many components have a preffered orientation, caused by its geometry, the pose spectrum can be restricted. This later limits the necessary variations of scene compositions required. Possible symmetry in shape can also help to reduced this parameter space. For possible physics-simulation the properties of the components with respect to flexibility have to be considered.

Viewing Properties

Depending on the vision system used for the task, multiple parameters have to be defined to generate training data. As some variational restrictions might arise from the 3D composition, a relational semantic has to be defined. In our use-case the type of delivery unit defines the placement of the camera. However, for different use-cases this might not be the case. E.g. for creation of training data out of the view of a robotic manipulator, more and inherently variant viewpoint variations have to be considered. Further as the visual set-up may differ between mono- and stereo-setups, field of view and similar parameters, these relations are defined in the scene grammar.

Background and Lighting Properties

Depending on the application, the background of the composed scene may be in the field of vision. In principle it is possible to use a parametrization and scene composition for the surrounding as well. Such might be needed for autonomous mobile robotics with variable camera handling, but lesser for static top-view settings. Therefore, randomized cluttered backgrounds similar to [4] may be utilized, if necessary.

As lighting is a highly variational and environmental dependent scene parameter, it can be emulated by randomized light sources of variant strengths. A relationship may occur if cameras are equipped with ring lights or similar.

3.4 Derivation of a Parameter Space

We now utilize the formulated grammar and derive a scene parameter space for the transportation of two components in boxes with varying packaging infill on a delivery unit with three levels. This setting is shown in Fig. 4: A pointer indicates the to be modelled tray setting. We assume camera settings on top of the box as indicated in the schematic. Each component is transported in its own box. Different box colours and fillings are predefined. For the filling packaging flips, bubble wrap and anti-slip mat are considered.

As one of the objects is equipped with a display and buttons on top, we assume a preferred orientation with the display upwards. For the variations with anti-slip mat and no in-fill, the pose of the objects are restricted to rotation around the vertical axis and translation in plane. For the fillings with packaging flips and bubble wrap, 6D poses are possible. However, to achieve randomized yet physic accurate 6D poses, a physics simulation is done. The objects are considered stiff, leading to rigid body simulation. The quantitative values of this parameter space are shown in Fig. 5a, b.

With this formulation of the use-case derived, a pipeline can be utilized to randomize the composition of scenes and render them.

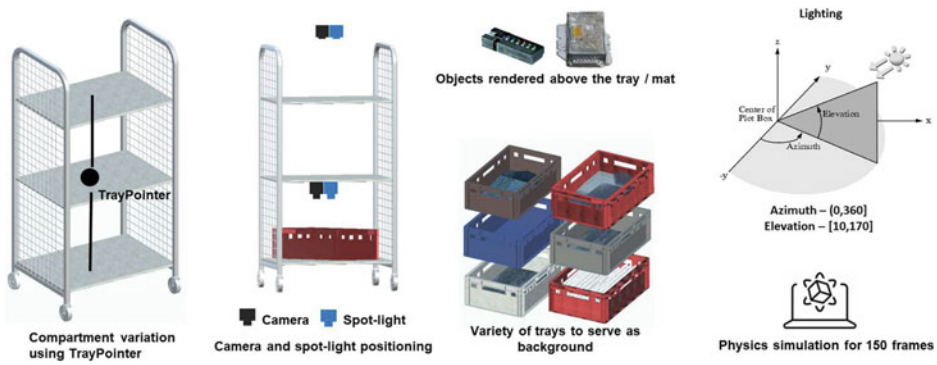
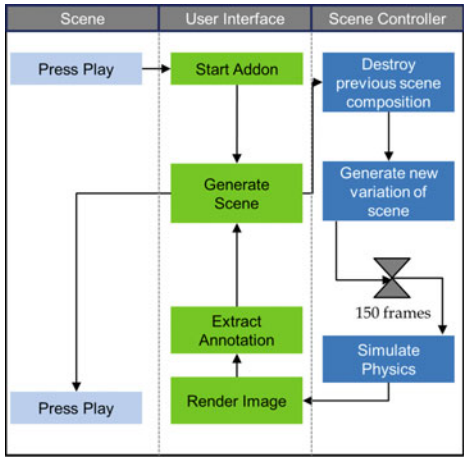


Fig. 4 Semantic interpretation and definition of variations according to the defined grammar



(a) Sequence Diagramm

Parameters		Variation
TrayPointer		Center of the 3 compartments
Tray	Combination	Empty Traiy, Bubble-wrap Packaging fills, and mats
	Position	X & Y-axis: ± 10 cm
	Orientation	A random rotation along Z-axis
	Colour	Random RGB colour
Objects	Number	Min: 0, Max: 1
	Position	X & Y-axis: ± 10 cm
	Orientation	Random Euler Angle
Light	Azimuth	0° and 360°
	Elevation	10° and 170°
Camera		60 cm above TrayPointer
Spot-Light	Location	60 cm above TrayPointer
	Intensity	0 to 8 (Unity quantification)

(b) Parametrization Table

Fig. 5 Sequence diagramm of the pipeline in (**a**), parametrization of the setting in (**b**)

4 Data Generation Pipeline

Implementation of the derived parameter space is done with the Unity Game Engine. We wrote an addon that generates scenes according to the defined parameter space. A sequence representation of that pipeline is shown in Fig. 5a. After generation and simulation of a scene variation, images are rendered and annotations extracted. In each loop, the parameters are changed according to the defined parameter space and Fig. 4. Quantitative formulation of the parameter space is shown in Table 5b.

Annotations are generated by our written Unity addon used for creation of the scene. Besides class of the picture, also the bounding box, and pose of the object is extracted and saved.

5 Validation

In order to validate that the scene grammar and implementation is capable of generating viable training data, we train a Deep-Learning image classifier with the synthetic images and test it against a hand-annotated real world data.

(a) Synthetic Training Data (b) Real World Captures

Fig. 6 Examples of generated synthetic training data (**a**) and real world data (**b**)

5.1 Model and Training

We use a ResNet34 architecture. The final classification layer consists of three classes, representing the Object 1, Object 2 and None Class. The model is trained in two stages. First the top fully connecte layers are trained, afterwards the entire model is trained. One cycle policy was used, with learning rate maximum 0.001 and minimum 0.0002. In both stages 5 epochs were trained. The pipeline generated 1000 synthetic images.

5.2 Results

We evaluate the trained network on for the network previously unknown data sets captured from a real word scene. This test set consists of 745 images with 329 *Object 1* images, 361 *Object 2* images and 55 *None* class Images. The confusion matrix of the results is shown in Table 1: ten out of the 745 images were miss-classified with 6 *Object 1* images not being detected and three being classified as *Object 2*. One *Object 2* image was classified as *Object 1*. This translates into a classification accuracy of 98.50%.

Table 1 Confusion matrix of the the real world classification test-set

		Predicted		
		Object 1	Object 2	None
Actual	Object 1	328	3	6
	Object 2	1	358	0
	None	0	0	49

5.3 Discussion

With few missclassifications, mostly concerning the object 1 class, the general classification task can be regarded successful. However, it is to be expected that more classes and a more challenging test set would have a negative impact on the results. In general, when training with synthetic data and applying that AI, the domain adaptation problem poses to open world capability of the AI. Missmatches between the synthetically created content and the real world as well as the difference between a rendered image and a real sensor perception, may prevent a successful use of the AI. In our case, with strict modelling of realistic transportation settings, we achieved to narrow the content gap between synthetic and real world for our use-case. As seen from Fig. 6, the real images contain blurring and glares, that are not accounted for in the synthetic images. Closing such appearance gaps is focus of different approaches, which could be combined with our developed semantics and pipeline but was out of scope for this work.

With the task of enabling an image classifier for the presented use-case being successfully fulfilled, the derived grammar can be considered a viable contribution to industry ready synthetic training data. However, we'd like to point out that the actual implementation of the parametrization space is not necessarily universal to every use-case. For example, the developed pipeline does not handle multi-object detection tasks and is as such less viable for vision systems with a greater field of view. Generalizing this implementation to include most of the grammars problem statements is aim for future work.

6 Conclusion and Outlook

In this work we aimed to enable an image classifier network for usage in intralogistic transport scenarios. To achieve this, we formulated a scene grammar for such scenarios and derived a parameter space for a given use-case. We implemented this use-case in a pipeline and utilized this to generate training data. This data is then used to train a Deep-Learning image classifier and validated against real world data.

Future work will focus on further generalization of the pipeline, to reduce the necessary transfer effort between formulating a scene parameter space and implementing it in a pipeline. Also a native integration with state of the art domain adaptation may be necessary, when used for more challenging tasks.

Acknowledgements Research was funded by the German Federal Ministry for Economics and Energy under the Program LuFo V-3 DEPOT.

Gefördert durch:

aufgrund eines Beschlusses
des Deutschen Bundestages

References

1. Sliwinski, M., Raabe, C.M., et al.: Modulare Ladungsträger für den Kleinteiletransport. ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb **115**, 418–21 (2020)
2. Jalal M, Spjut J, et al. SIDOD: A synthetic image dataset for 3D object pose recognition with distractors. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. Vol. 2019-June. 2019:475–7. https://doi.org/10.1109/CVPRW.2019.00063
3. Bousmalis K, Irpan A, et al. Using Simulation and Domain Adaptation to Improve Efficiency of Deep Robotic Grasping. In: 2018 IEEE International Conference on Robotics and Automation (ICRA):4243–50. https://doi.org/10.1109/ICRA.2018.8460875
4. Hinterstoisser S, Pauly O, et al. An Annotation Saved is an Annotation Earned: Using Fully Synthetic Training for Object Instance Detection. In: 2019 IEEE International Conference on Computer Vision Workshop (ICCVW)
5. Kar A, Prakash A, et al. Meta-sim: Learning to generate synthetic datasets. In: Proceedings of the IEEE International Conference on Computer Vision. 2019:4550–9. https://doi.org/10.1109/ICCV.2019.00465
6. Gaidon A, Wang Q, et al. VirtualWorlds as Proxy for Multi-object Tracking Analysis. In: 29th IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, 2016:4340–9. https://doi.org/10.1109/CVPR.2016.470
7. Tremblay J, To T, et al. Deep Object Pose Estimation for Semantic Robotic Grasping of Household Objects. In: Conference on Robot Learning. 2018
8. Kleeberger K, Landgraf C, and Huber MF. Large-scale 6D Object Pose Estimation Dataset for Industrial Bin-Picking. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2019:2573–8. https://doi.org/10.1109/IROS40897.2019.8967594
9. Brucker M, Durner M, et al. 6DoF Pose Estimation for Industrial Manipulation Based on Synthetic Data. In: Proceedings of the 2018 International Symposium on Experimental Robotics. Vol. 11. 2020:675–84. https://doi.org/10.1007/978-3-030-33950-058
10. Andulkar M, Hodapp J, et al. Training CNNs from Synthetic Data for Part Handling in Industrial Environments. In: IEEE International Conference on Automation Science and Engineering. Vol. 2018-August. 2018:624–9. https://doi.org/10.1109/COASE.2018.8560470
11. Lin TY, Maire M, et al. Microsoft COCO: Common Objects in Context. In: Lecture Notes in Computer Science book series. Vol. 8693:740–55. https://doi.org/10.1007/978-3-319-10602-148
12. Dahmen, T., Trampert, P., et al.: Digital reality: a model-based approach to supervised learning from synthetic data. AI Perspectives **1**, 1–12 (2019)
13. Tobin J, Fong R, et al. Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2017. https://doi.org/10.1109/IROS.2017.8202133