Bridging the Reality Gap: Synthetic Data Generation for Food Portion Estimation

Ben Gur
Stanford University
Department of Computer Science
bgur@stanford.edu

Abstract

Artificial intelligence and deep learning is founded on data. The ability to generate synthetic data is a transformative technique for enabling the creation of large-scale, diverse datasets without the constraints of manual aggregation and labeling. This project explores the application of synthetic data generation through the application for food amount estimation, pulling existing research/applications of the "digital twin" concept. It includes a comprehensive synthetic data generation pipeline with physics-based simulations through the Genesis Engine, utilizing methods like domain randomization to bridge the reality gap. The YOLOv8 model was used with a custom regression head for weight estimation, trained with synthetic, real and hybrid data. After various experimentation this project demonstrates the viability of synthetic data for food portion estimation and compares different training approaches for achieving better performance. One surprising discovery was an ablation study revealing that our simplest loss function outperforms researched weighting strategies for uncertainty, prompting questions relating to multi-task learning. This project's findings provide significant implications for synthetic data applications and insights on model design.

1. Introduction

With the immovable importance of data in computer learning, data needs and aggregation challenges are a constant bottleneck and barrier in developing models, especially those with computer vision components. The process of collecting, labeling and validating large-scale datasets is often time consuming, expensive and widely impractical in both common and specialized areas. Synthetic data generation offers a promising alternative, enabling the creation of virtually infinite training examples with annotations at a fraction of the resources, given the effective development of a generation pipeline.

"Digital twin" is often used to refer to the replication of physical objects or processes and has gained significant attention in its practical uses today. From training robotic systems [5] to simulating protein folding and materials science, to creating realistic traffic models for autonomous vehicles [6], synthetic data has demonstrated remarkable utility in bridging the gap between simulation and reality.

Food portion estimation is used in this project as an effective task for synthetic data applications. Simulating food images provides an opportunity to create a controlled environment with great iterative potential and utilize comparisons between generated and real data. Although tools already exist for this task, it also doubles as an exciting application since it relates to areas important to most healthy people such as nutrition tracking, dietary management, and food waste reduction. Typically the downside in this task is collecting diverse, accurately labeled food data that provides enough variability in appearance, portions, angles, lighting, etc.

The primary challenge synthetic data faces is "bridging the gap" and ensuring models can generalize to the real world effectively. One way to support this bridging is by utilizing physics-based simulations with automated domain randomization to capture the before-mentioned variability.

In this report our key contributions include:

- A scalable synthetic data generation pipeline using the Genesis physics engine, capable of producing thousands of diverse food images with precise weight annotations
- Effective domain randomization strategies for food simulation, including variations in lighting, camera parameters, food distribution, and plate characteristics
- Novel application of YOLOv8 for weight estimation along with it's trained ability for food recognition
- Findings on synthetic-to-real transfer, demonstrating the viability of synthetic data for real world tasks

- Unexpected insights on loss function design for weight regression, providing insights on common expectations about uncertainty weighting in multi-task learning

This work mainly demonstrates that synthetic data can effectively supplement and one day replace real data for specialized computer vision tasks.

2. Related Work

2.1. Synthetic Data Generation for Computer Vision

The use of synthetic data in computer vision has evolved significantly in recent years, driven by advances in rendering technology and physics-based simulations. The digital twin concept has found very interesting and useful applications across various domains.

In this new use of synthetic data, randomization has emerged as a great technique for bridging the reality gap. Tobin et al. [5] demonstrated that by randomizing non-essential aspects of a simulated environment (lighting, textures, camera positions), models could learn to focus on the essential features of a task, enabling effective transfer without requiring photorealistic rendering. Tremblay et al. [6] extended this approach to object detection tasks, showing that models trained on synthetic data with domain randomization could achieve 95% of the performance of models trained on real data.

Physics-based simulation approaches have been particularly successful in robotics and autonomous driving, though accurate modeling of physical interactions is crucial. These approaches have enabled training of models for tasks that would be impractical or dangerous to learn in the real world, such as autonomous vehicles learning to drive on public roads without causing accidents.

Success stories with synthetic data cover a wide range of domains, including robotic training, autonomous vehicles, and medical imaging. For example, synthetic data has been used to train models for detecting rare medical conditions where real training data is scarce.

2.2. Food Recognition and Portion Estimation

Food portion estimation presents unique challenges for computer vision systems. Traditional approaches relied on geometric approximations and reference objects for scale, while deep learning approaches have enabled more direct estimation from images.

Min et al. [3] provided a comprehensive study of food computing, and highlights the challenges in data collection and annotation for food recognition tasks. The inherent diversity of food appearance, portion sizes, lighting and camera angles makes creating large-scale, diverse datasets difficult.

Deep learning approaches have shown promising results for food recognition, but portion estimation is still challenging due to most approaches relying heavily on extremely large sets of real-world data with precise weight labelling.

2.3. Loss Function Design and Multi-Task Learning

Loss function design plays a crucial role in training deep learning models, particularly when a model is trained for multiple tasks. Kendall et al. [2] discusses an approach to multi-task learning using uncertainty to weigh different loss components. The method interprets task-specific uncertainty as a measure of the relative importance of each task, allowing the model to learn better weights in training.

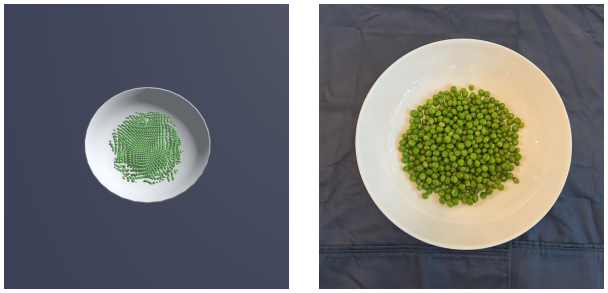
Intuition and research with task uncertainty often points to it outperforming fixed weighting by adapting the relative difficulty of different but adjacent tasks. Experiments conducted in this project with different weighing strategies have shown mixed results however.

Multi-task learning in computer vision typically involves shared feature representations across related tasks. The design of task-specific heads and the sharing of features between them can significantly impact performance, often for the better. Understanding the balance between task-specific and shared representations is an active area of research today.

3. Data

3.1. Synthetic Dataset Generation

The synthetic data generation pipeline leverages the Genesis physics engine [4] to create realistic simulations of food portions on plates. The pipeline consists of several key components:



Synthetic Data Real Data

Figure 1. Comparison between synthetic data (left) and real data (right) used for training the food portion estimation model.

3.1.1 Physics-Based Modeling

Peas are modeled as a granular food type using the MPMEntity with Sand particle materials in Genesis, which seemingly captures the physical behavior of small, round

food items with some effect. A lot of time was spent calibrating physics parameters to match the density and behavior of real peas, ensuring that the relationship between visual appearance and weight was preserved in the generation.

For each weight category (50g, 150g, 300g, 450g, 600g), we calculated the appropriate number of pea particles based on the average weight of individual peas, promoting physical accuracy. This required developing custom code to calculate accurate pea counts for different weights and applying it with the scene generation options available in Genesis, which was challenging.

3.1.2 Implementation Challenges

Implementing the synthetic data generation pipeline presented numerous challenges. The Genesis engine, while powerful, has a steep learning curve and required significant effort to configure properly for even simple simulations. Generating accurate representations of peas required writing a good deal of custom code and manually tuning physics parameters.

Setting up a working domain randomization framework also required careful design to ensure that the randomization parameters were meaningful and covered the needed range of variation without generating useless data. This process involved a lot of iterative work involving visual inspection of the generated images and comparison with real-world examples.

3.1.3 Dataset Statistics

The final synthetic dataset consisted of 9,750 images with the following characteristics:

- **Weight Categories**: 50g, 150g, 300g, 450g, 600g (evenly distributed)
- **Lighting Conditions**: cool daylight, warm, studio lighting, overcast_blue, and 6 others (evenly distributed)
- **Food Configurations**: centered, off-center, scattered, narrowed, spread (evenly distributed)
- **Camera Angles**: front, back, left, right, top (evenly distributed)
- **Camera Heights**: high, mid, low (evenly distributed)
- **Resolution**: 640x640 pixels

Each image was automatically annotated in a paired .txt file with bounding box data and the corresponding weight value, eliminating the need for manual annotation.

3.2. Real-World Dataset

To complement the synthetic data and provide a basis for evaluation, we collected a real-world dataset of 100 pea images:

- **Weight Categories**: 50g, 150g, 300g, 600g
- **Configurations**: centered, spread, off-center, two piles, scattered
- **Camera Angles**: front, left, back, right, top
- **Resolution**: 640x640 pixels (resized from 3024x3024)

The real data collection process was significantly more time-consuming than synthetic data generation (not including development), requiring careful measurement of food weights, consistent camera positioning, and manual annotation of images. This highlights one of the key advantages of synthetic data mentioned: the ability to generate large amounts of perfectly annotated data with minimal effort.

3.3. Combined Hybrid Dataset

For our hybrid training approach, we combined the synthetic and real datasets:

- **Training Split**: 70% (real) + 70% (synthetic)
- **Validation Split**: 15% (real) + 15% (synthetic)
- **Test Split**: 15% (real) + 15% (synthetic)

All images were annotated for use with YOLOv8 with an additional weight value, allowing for simultaneous training of detection and regression tasks for added experimentation with multi-task learning and uncertainty. The complementary nature of real and synthetic data provided a rich training set that combined the diversity and scale of synthetic data with some transferred authenticity from real-world examples.

4. Methods

4.1. Synthetic Data Generation Methodology

Our synthetic data generation methodology focused on creating physically accurate simulations of pea portions that would transfer effectively. The key aspects of the methods used are:

Physics-Based Simulation: By using the Genesis physics engine to simulate the behavior of peas on plates, the physical properties (mass, density, friction) matched real-world values and provided accurate physical behavior.

Accurate Weight-to-Appearance Mapping: For each weight category, the appropriate number of pea particles was calculated based on the average weight of individual

peas and behavior of MPMEntities in genesis, ensuring that the generated particles representing food accurately reflected specific amounts.

Domain Randomization Implementation: As mentioned a comprehensive domain randomization framework was built to automatically vary lighting conditions, camera parameters, food distribution patterns, and plate characteristics.

Automated Generation Pipeline: The entire pipeline was automated through a series of scripts that generated thousands of diverse images with labelling. This pipeline could be easily scaled to generate additional data as needed, with added variability and photorealism as a fraction of the available functionality in genesis was learned and utilized due to resource constraints.

4.2. Model Architecture

Model architecture was built ontop of YOLOv8 [1], a state-of-the-art object detection framework for image recognition:

YOLOv8: YOLOv8m provided a good balance between performance and computational efficiency. The backbone consists of a series of convolutional layers with blocks for feature extraction.

Weight Regression Head: We added a custom regression head that utilizes features from the existing model and predicts the weight of the detected food item. This head consists of several convolutional layers followed by a pooling layer and a fully connected layer that outputs a weight value.

Feature Sharing: The pre-existing detection head and the custom regression head share the same features, allowing for efficient multi-task learning and sharing of common visual features thought to be relevant to both tasks.

4.3. Loss Function Design

Our loss function combined the available detection and regression predicting:

Detection Loss: The pre-existing YOLOv8 detection loss, which includes components for classification and localization.

Regression Loss: Mean Squared Error (MSE) loss between the predicted and labeled amounts.

Combined Loss: The detection and regression losses were combined using one of three uncertainty weighting strategies:

- **Learned Uncertainty Weighting**: Based on Kendall et al. [2], this approach has task-specific uncertainty that automatically adjusts for the detection and regression losses during training.
- **Fixed Uncertainty Weighting**: Equal weights (1.0) for both detection and regression tasks.
- **No Uncertainty Weighting**: No explicit weighting between tasks, simply added losses together.

4.4. Training Strategy

We employed a three-stage progressive training approach:

Stage 1: Train only the detection head, freezing the regression head.

Stage 2: Train only the regression head, freezing the detection head.

Stage 3: Joint training of both heads.

This progressive approach allowed each task to established independent learning before joint training, the worry being that one task (namely the pre-existing detection) could dominate the other during training.

As an experiment three models were trained:

- **Synthetic → Real**: Trained on synthetic data, evaluated on real data
- **Real → Real**: Trained on real data, evaluated on real data
- **Hybrid → Real**: Trained on combined synthetic and real data, evaluated on real data

For each model, we used Adam optimization with a learning rate of 0.001 and a batch size of 4.

5. Experiments

5.1. Synthetic Data Viability Analysis

The performance of models trained on synthetic, real, and hybrid datasets when evaluated on real-world test data for comparison:

Synthetic → Real Transfer: The model trained only on synthetic data achieved a Mean Absolute Error (MAE) of 980.23g and 0.00% of estimates within 20% of the true weight. While performance could likely be improved vastly with better simulations, it demonstrated the difficulty in learning meaningful features from synthetic data.

Reality Gap Assessment: The gap between synthetic and real data training (980.23g vs. 253.44g MAE) indicated that there are still significant differences between our synthetic and real data distributions. However, the hybrid model's improved performance suggested that synthetic data could effectively complement real data.

5.2. Baseline Experiments

Our baseline experiments established the performance of different training approaches:

Table 1. Baseline Experiment Results

Experiment	MAE (g)	RMSE (g)	Within 20% (%)
Synthetic → Real	980.23	1024.25	0.00
Real → Real	253.44	320.50	0.00
Hybrid → Real	681.22	931.16	24.49

Synthetic → Real: Training on synthetic data only resulted in high error rates did not effectively demonstrate an ability to capture the overall relationship between visual features and weight.

Real → Real: Training on real data only achieved better MAE (253.44g) but surprisingly poor performance, likely because the model was overfitting to the small real dataset.

Hybrid → Real: Training on combined synthetic and real data achieved the best overall performance, with 24.49% of estimates within 20% of the true weight indicating solid understanding of the amount-appearance relationship. Increased epochs in later experiments focused on improving hybrid data use were able to achieve 40.00% of estimates within 20% of the true weight.

5.3. Ablation Study: Uncertainty Weighting

We conducted an ablation study to evaluate the impact of different uncertainty weighting techniques on performance:

Table 2. Uncertainty Weighting Ablation Results

Strategy	MAE (g)	RMSE (g)	Within 20% (%)
Learned	355.74	391.17	0.00
Fixed	125.87	189.27	20.00
None	116.41	160.42	33.33

Learned Uncertainty Weighting: Surprisingly, the researched approach performed worst, with an MAE of 355.74g and 0.00% of estimates within 20% of the true weight.

Fixed Uncertainty Weighting: Equal weighting of tasks performed better, with an MAE of 125.87g and 20.00% of estimates within 20%.

No Uncertainty Weighting: The simplest approach performed best, with an MAE of 116.41g and 33.33% of estimates within 20%.

These results contradicted the theoretical expectation that learned uncertainty would adaptively find the optimal weight between task components. Instead, they suggested that for this specific experiment, the detection and regression heads naturally balanced each other without dynamic weighting.

5.4. Performance Analysis

5.4.1 Detailed Error Analysis

Analysis of the predictions from the best model (No Uncertainty Weighting) revealed interesting patterns:

Performance by Weight Category:

- **50g Category**: 50% of samples within 20% of true weight
- **150g Category**: 25% of samples within 20% of true weight
- **300g Category**: 50% of samples within 20% of true weight
- **600g Category**: 0% of samples within 20% of true weight

Performance by Food Configuration:

- **Centered**: Generally moderate performance (40-58% error)
- **Off-center**: Highly mixed performance (11-84% error)
- **Scattered**: Highly mixed performance (3-77% error)
- **Two piles**: Moderate performance (16-53% error)
- **Spread**: Generally poor performance (53-212% error)

Performance by Camera Angle:

- **Front**: Moderate performance (3-53% error)
- **Top**: Highly mixed performance (16-84% error)
- **Side views**: Extremely mixed performance (18-212% error)

Weight Estimation Bias: The model tended to underestimate larger amounts (600g) and overestimate smaller amounts (50g), possibly due to a regression towards the mean or just inaccuracy in synthetic data.

5.4.2 Qualitative Results

Visual inspection of the model's predictions revealed several insights:

Best Case Performance: The model achieved as low as 3.06% error in ideal conditions (300g, scattered, front view).

Worst Case Performance: Spread configurations led to errors exceeding 200%.

5.5. Computational Considerations

Synthetic vs. Real Data Collection: Synthetic data generation was significantly more efficient than real data collection once the pipeline was established. While the initial setup required substantial effort, subsequent data generation could be fully automated and scaled to produce thousands of diverse images with perfect annotations.

Training Approaches: Hybrid training required more computational resources than either synthetic-only or real-only training likely due to the combined data size, but also provided the best overall performance. The three-stage progressive training approach also added computation time but is thought to have improved performance.

Practical Implementation: Based on our experiments, we recommend a hybrid training approach with no uncertainty weighting for optimal performance. This approach balances the benefits of synthetic and real data while using a simple, effective loss function design.

6. Conclusion

6.1. Key Findings

Our research has demonstrated the viability of synthetic data for food portion estimation, with several key findings:

Synthetic Data Viability: While synthetic data alone was not sufficient for effective performance at current levels of simulation ability, it provided valuable complementary information when combined with real data. The hybrid approach achieved significantly better performance than synthetic-only training. It achieved higher performance than real-only as well though not much can be drawn due to the size of the data set.

Effectiveness of Hybrid Training: The combination of synthetic and real data is thought to have created a model that captured both the diversity of synthetic data and the authenticity of real examples, resulting in 40.00% of estimates within 20% of the true amount with additional epochs.

Surprising Loss Function Results: The simplest loss function design (no uncertainty weighting) outperformed more complex approaches, achieving an 82.9% reduction in MAE compared to the original hybrid model.

6.2. Implications for Synthetic Data Applications

Our work has several implications for synthetic data applications:

Digital Twin Potential: The success of our approach demonstrates the potential of digital twins for computer vision tasks, particularly in domains where real data collection is challenging or expensive.

Scalability Advantages: Once established, synthetic data generation pipelines offer significant scalability advantages over real data collection, enabling the creation of large, diverse datasets with accurate annotations.

Domain Randomization Effectiveness: Our results promote the effectiveness of domain randomization for bridging the reality gap, even without intense photorealistic rendering.

6.3. Implications for Model Design

Our findings also have implications for model design principles:

Simpler Can Be Better: The improved performance of the simplest loss function design challenges the assumption that more complex approaches are inherently better.

Task-Specific Considerations: The optimal approach may depend on the specific characteristics of the tasks being combined, suggesting that general principles should be applied with caution.

6.4. Limitations and Future Work

Following this project several opportunities for future work are presented:

Advanced Synthetic Data Generation: Future work could explore more sophisticated synthetic data generation techniques, including:

- Higher photorealism settings in Genesis
- More sophisticated lighting models (multidirectional, reflections)
- Greater variation in food properties (color, shape, texture)
- Expanded environmental diversity (backgrounds, surfaces, containers)

Model Architecture Exploration: Further exploration of regression head designs and feature sharing strategies could potentially improve performance.

Extended Training: Increasing the number of epochs with the optimal configuration would likely yield even better results.

Compute Time and Resources: With the steep learning curve and resource requirements for generation with Genesis, the typical limitation of time and resources for this project was compounded.

In conclusion, our work demonstrates the potential of synthetic data for food portion estimation while providing a few minor insights into model design. The combination of physics-based simulation, domain randomization, and experiment validated model design offers a promising approach for addressing data needs and challenges in computer vision applications.

References

[1] G. Jocher, A. Chaurasia, and J. Qiu. Yolov8: A sota model for object detection and image segmentation. *Ultralytics*, 2023.

[2] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7482–7491, 2018.

[3] W. Min, S. Jiang, L. Liu, Y. Rui, and R. Jain. A survey on food computing. *ACM Computing Surveys*, 52(5):1–36, 2019.

[4] G. Team. Genesis: A physics-based simulation engine for synthetic data generation. Genesis Documentation, 2023.

[5] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 23–30, 2017.

[6] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, and S. Birchfield. Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 969–977, 2018.