

Practical Methods to Estimate Fabric Mechanics from Metadata - Supplementary Material

H. Dominguez-Elvira^{1,2} and A. Nicas¹ and G. Cirio¹ and A. Rodriguez¹ and E. Garces^{1,2}

¹SEDDI, Madrid, Spain

²Universidad Rey Juan Carlos, Madrid, Spain

1. Model Implementation Details

For training both models, MECHMET and MECHIM, we used the Mean Absolute Error of the parameters as the loss function, and a different set of hyper-parameters for the *bending* and *stretch* models. For MECHMET, which is a Random Forest Regressor (RFR), we did a 5-fold cross validation to find, for stretch: $\text{min_samples_leaf}=0.0148$, $\text{min_samples_split}=0.0455$, $\text{max_depth}=10$, $\text{n_estimators}=100$, $\text{max_features}=0.3$. For bending: $\text{min_samples_leaf}=0.0045$, $\text{min_samples_split}=0.0055$, $\text{max_depth}=10$, $\text{n_estimators}=100$, $\text{max_features}=0.3$. Training both models takes a few seconds. For MECHIM, we use a ResNET18 [HZRS15] as feature extractor and the following sizes for the architecture components. For stretch: MLP layers=2, with an $\text{input_size}=\text{metada_size}*\text{positional_encoding} + \text{FC_size} + \text{composition_embedding_size}$. The size of the layers was progressively halved until reaching the last layer, which has an output size of 3 (the three stretch parameters). For bending, MLP layers=5, and the input_size and size of the layers follows the same pattern as the MLP for stretch. The FC layer size is 5 for both models. The compositions were reduced using an MLP with 5 layers for stretch and 4 layers for bending. The size of the layers was also progressively halved until reaching the last layer, which has a size of 2. We used an Adam optimizer with learning rates of 0.001 and 0.08 for stretch and bending models, and a learning rate decay of 0.78 for both models. For training the neural network we used crops of the images of size 256x256 and performed data augmentation by random perturbations of small rotations, Gaussian blurs, and changes of saturation, contrast, and color. The training was done with a GeForce RTX 2080 Ti during approximately 150 epochs (about 20 minutes).

2. Cusick Results

In this section we show additional results of the quantitative and qualitative performance of our models in the Cusick metric space.

2.1. Cusick Quantitative Results

In Figure 1 we evaluate the error in the Cusick metrics per family. As explained in the main document, Twill and Plain families have the highest variability of errors, because they group together

smaller categories of fabrics. Both models struggle with families that are underrepresented in our dataset, like Crepe or Fleece.

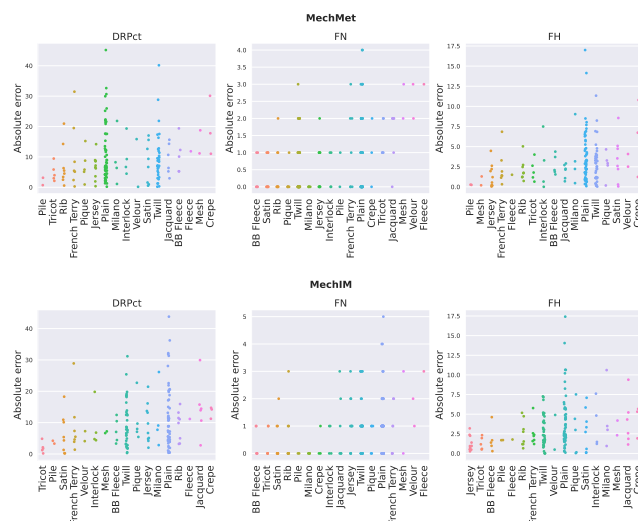
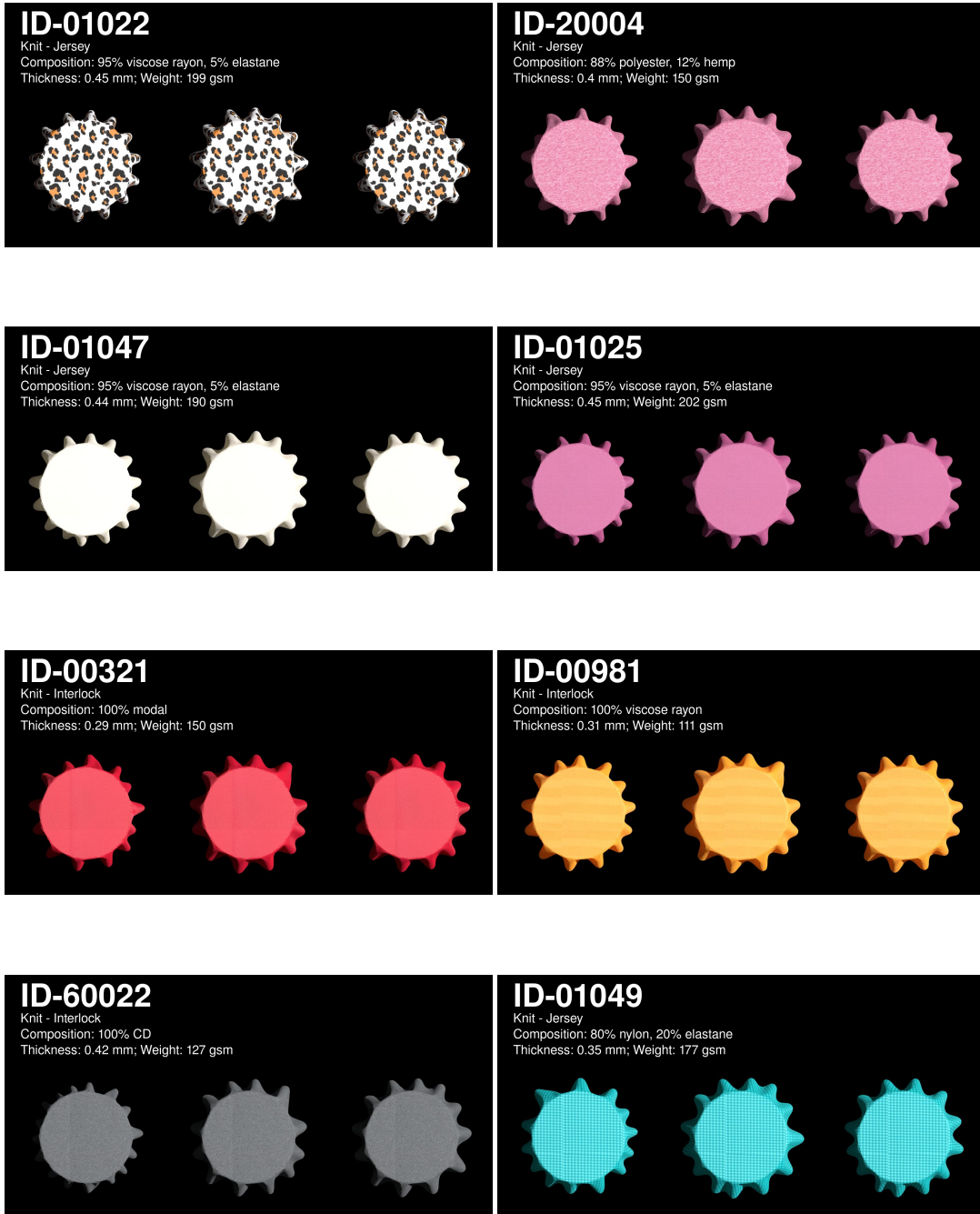
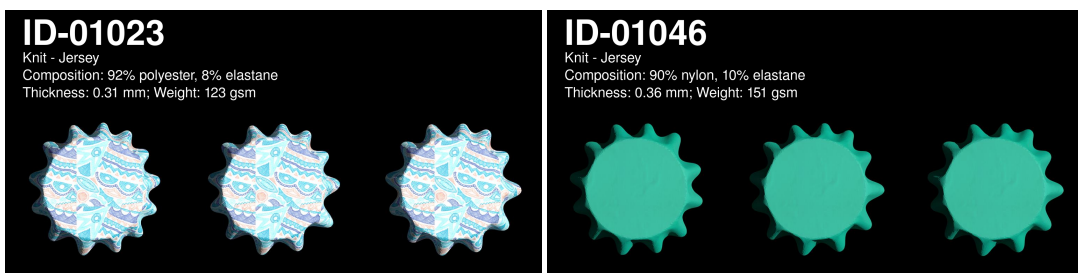


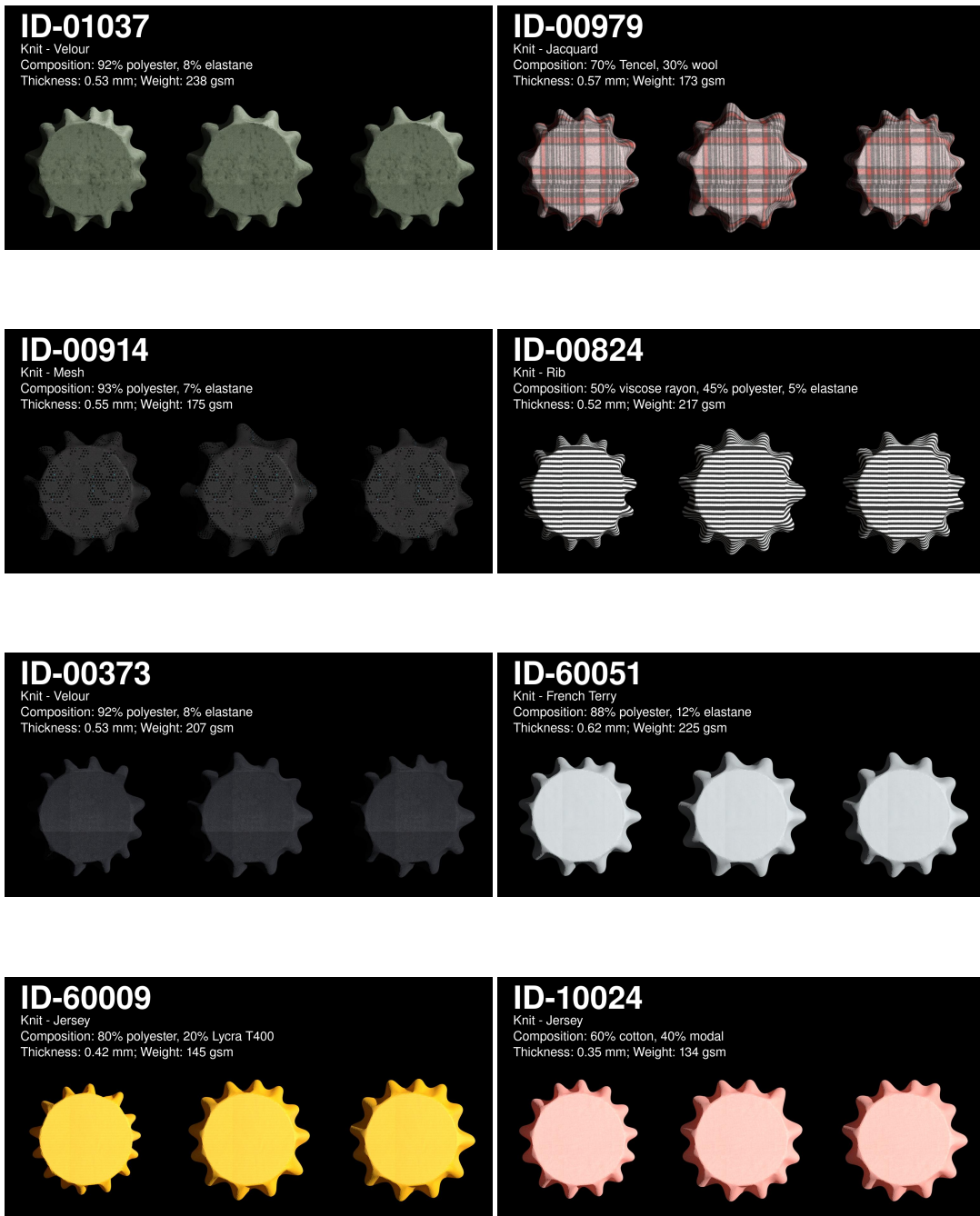
Figure 1: Absolute errors of DRPct, FN and FH of all the fabrics per family, sorted by increasing mean family error.

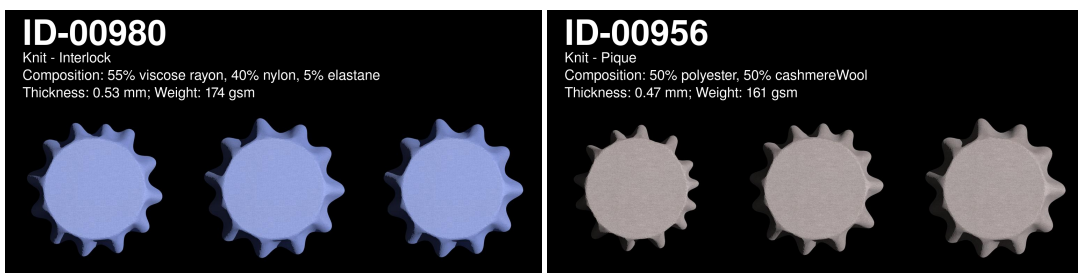
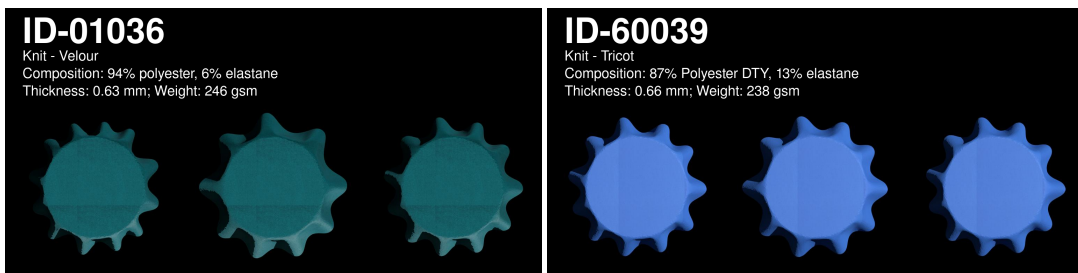
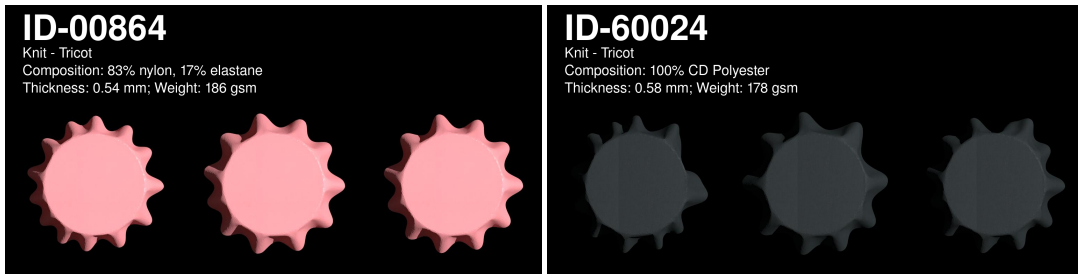
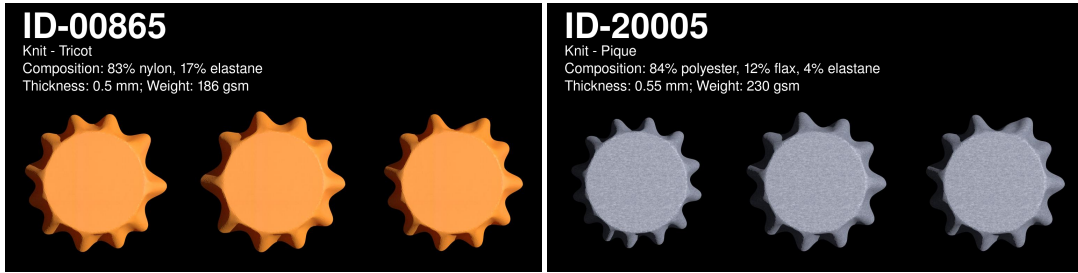
2.2. Cusick Qualitative Results

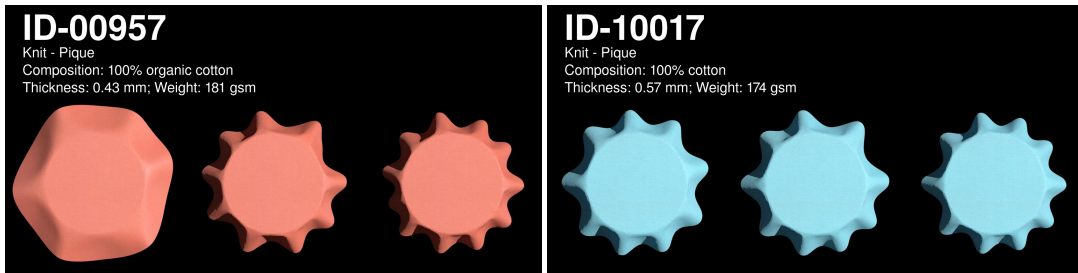
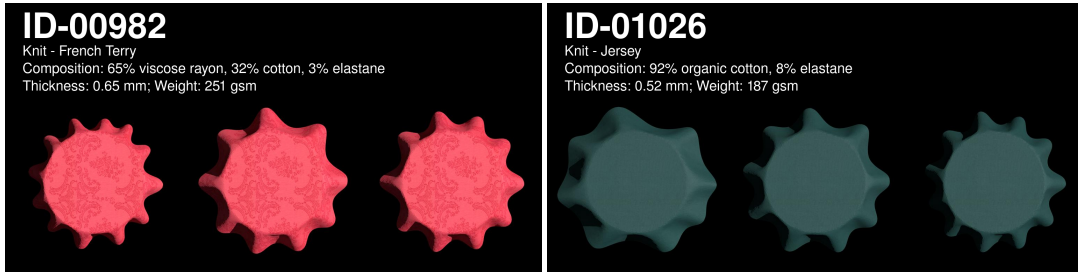
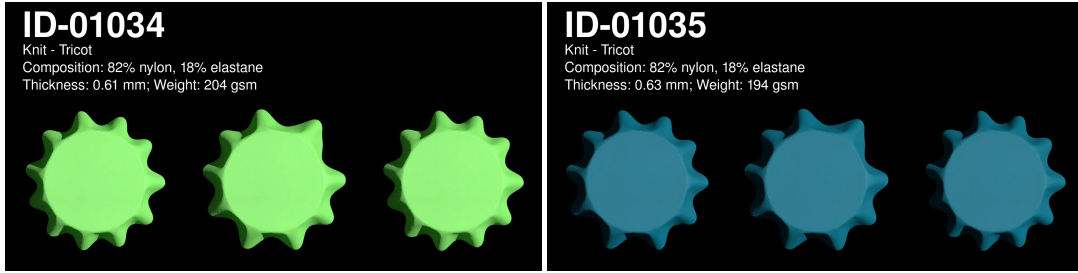
In the following pages we show a set of digitized fabrics from our test set sorted by the error of the Drape Coefficient (DRPcte) in increasing order. Each figure contains drapes of the Ground Truth (left), the MECHMET model (middle), and the MECHIM model (right).

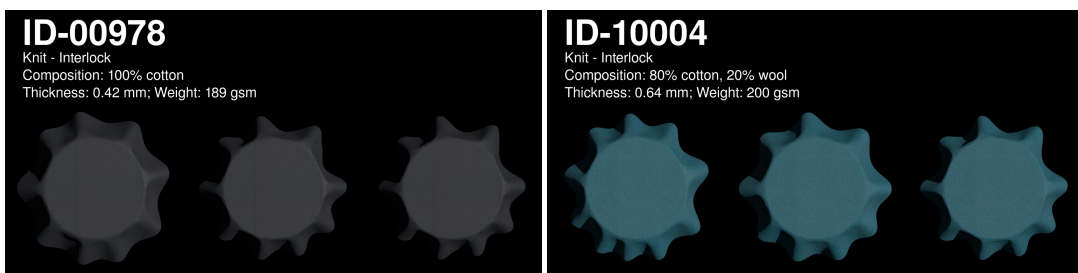
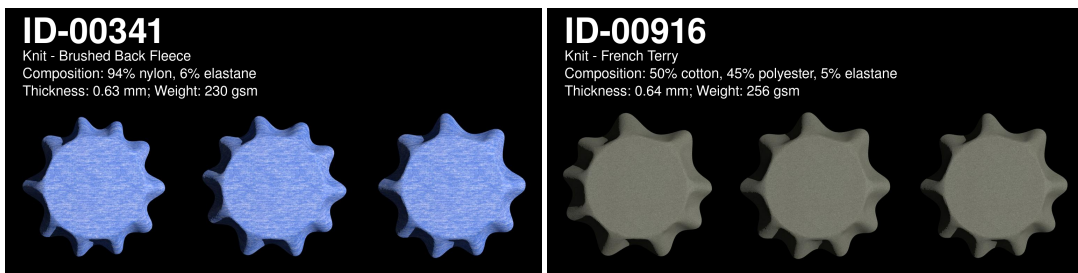
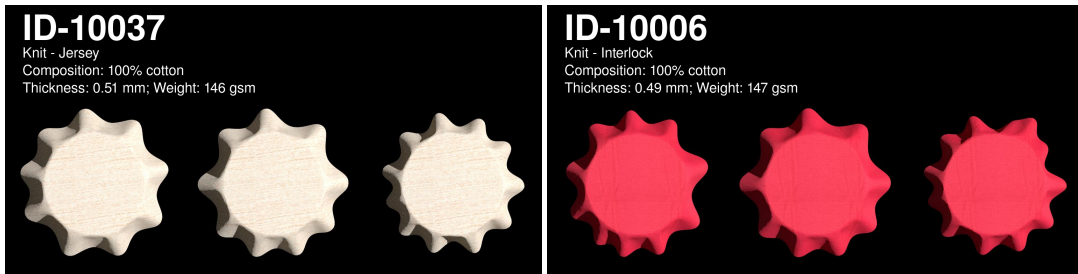
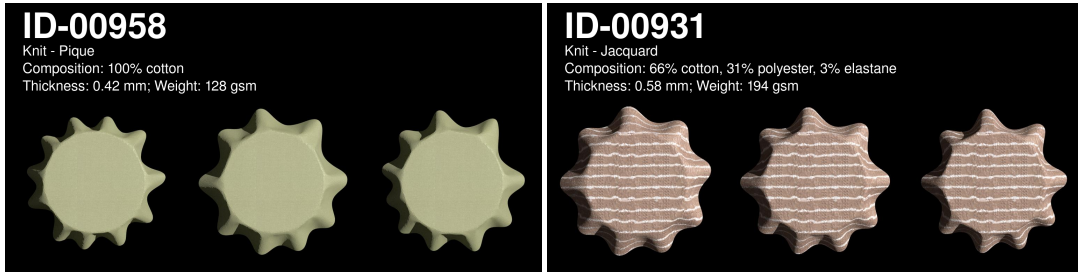


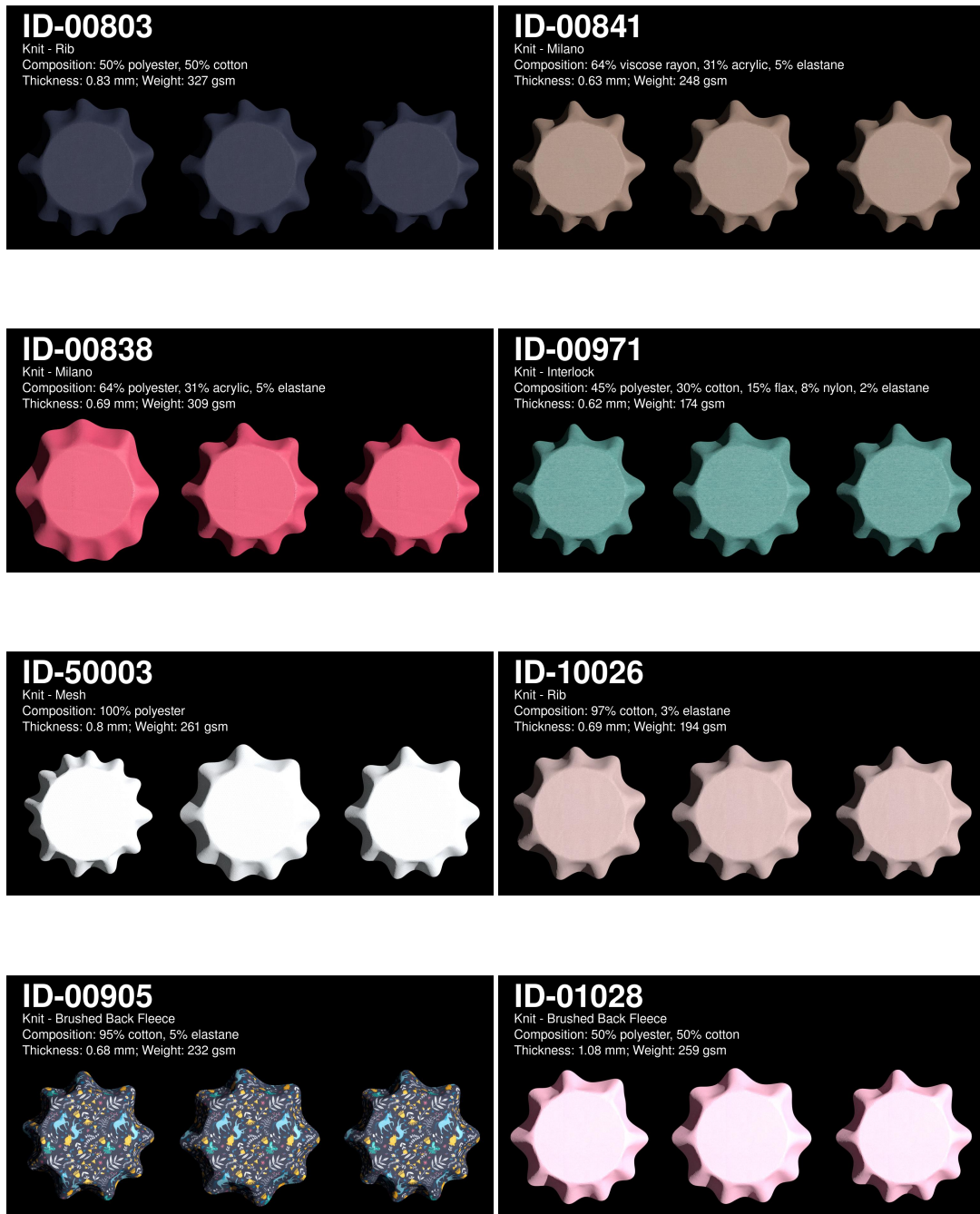


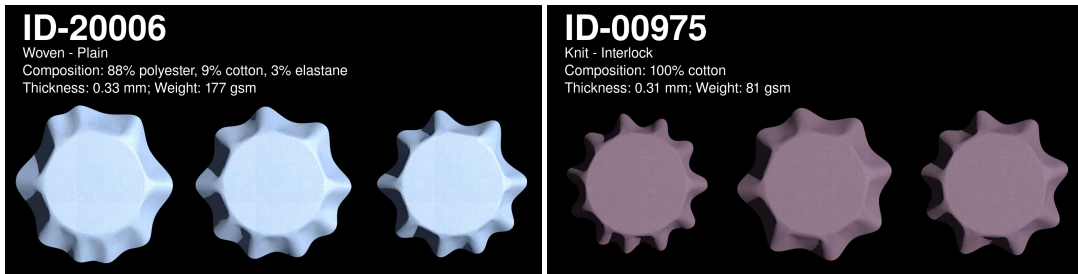
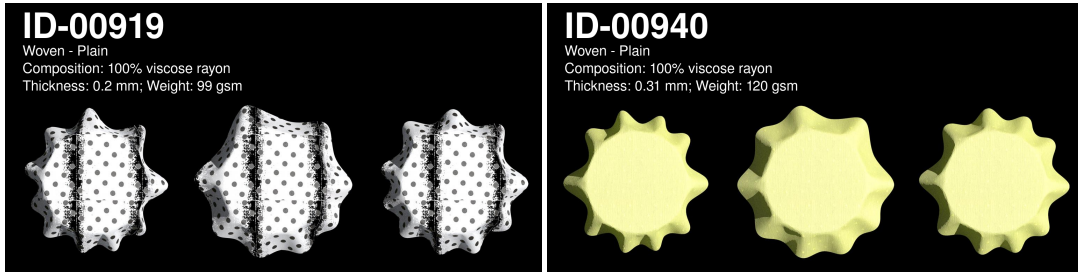


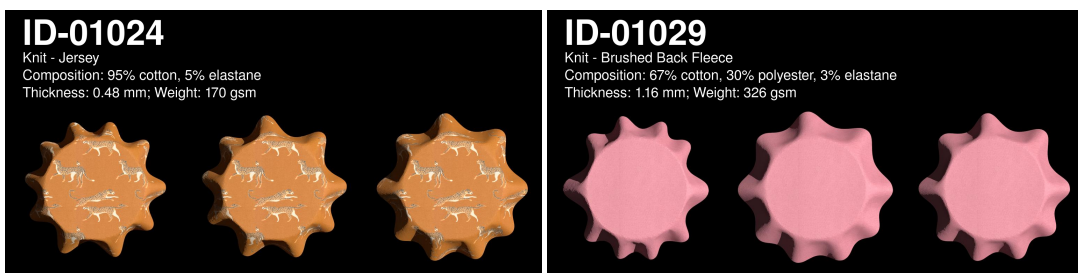
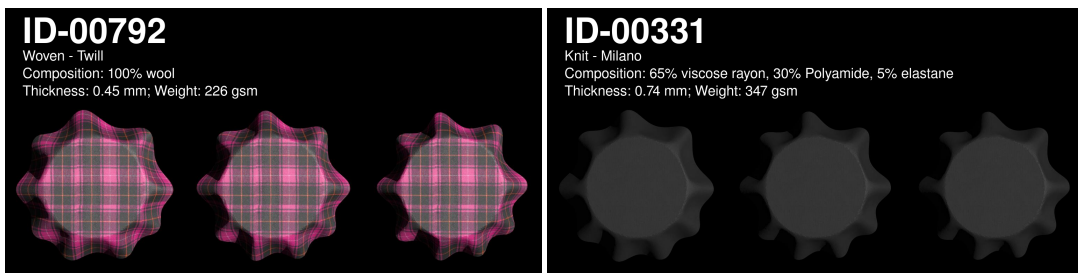
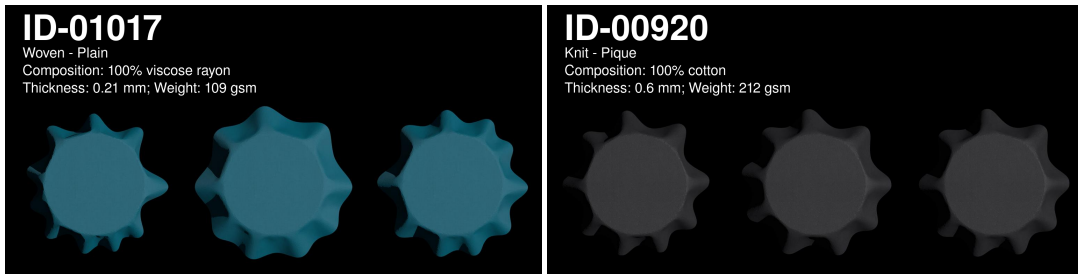
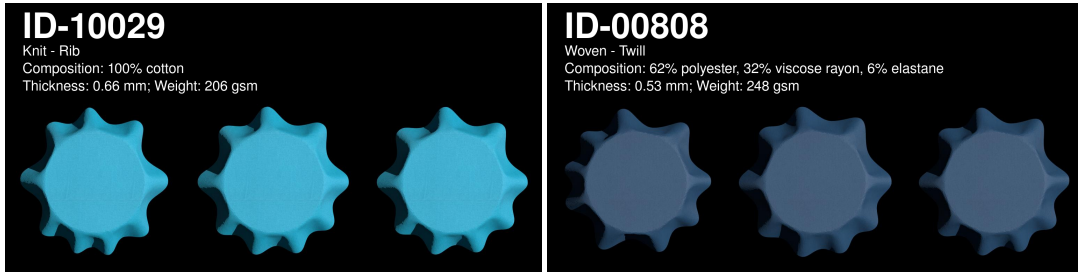


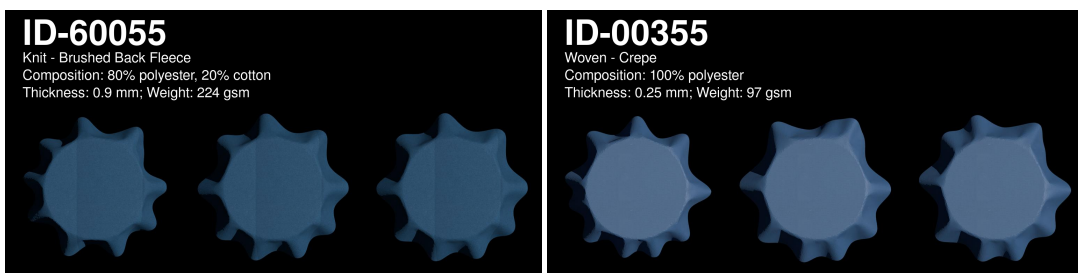
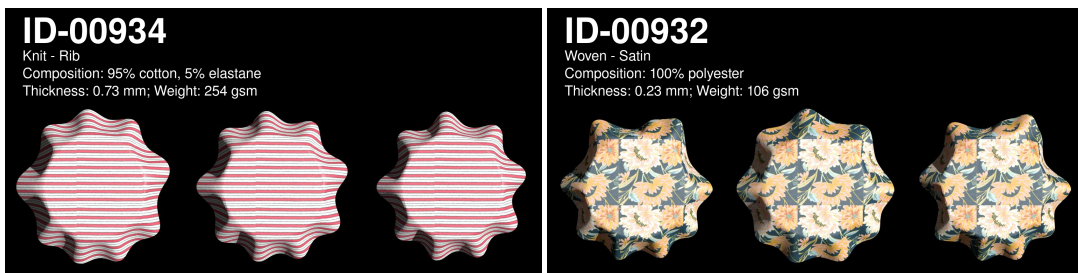
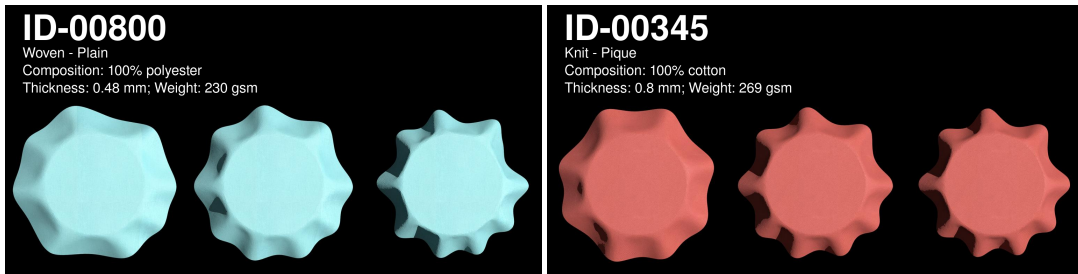
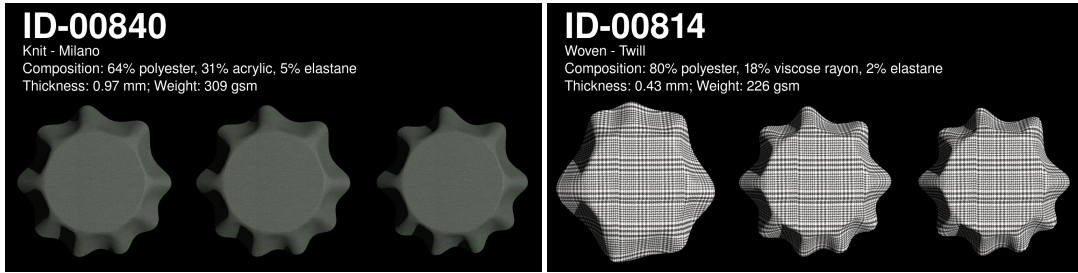


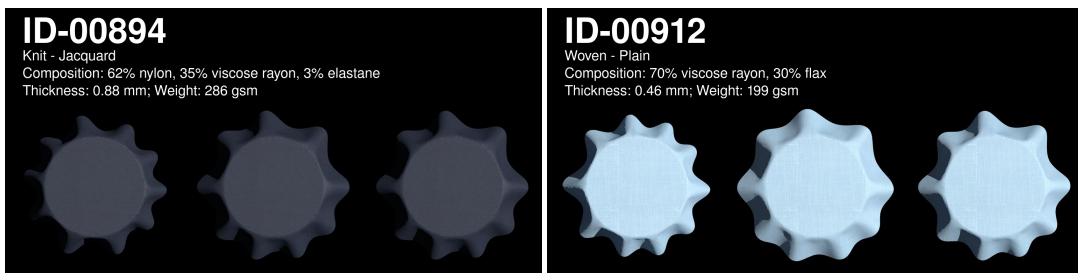
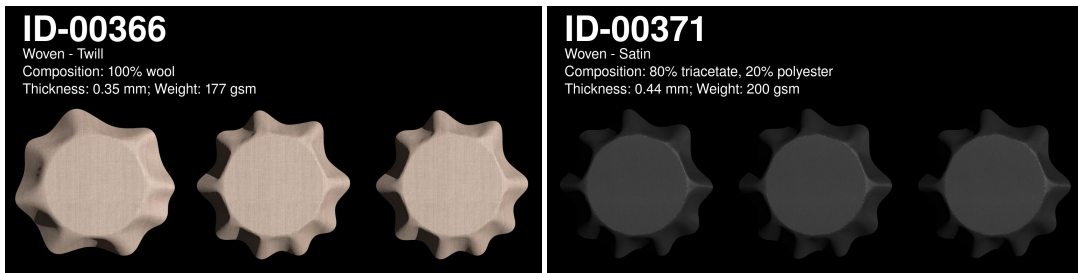
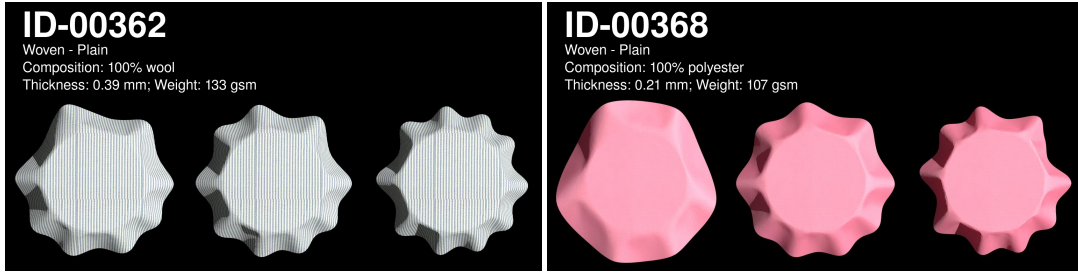
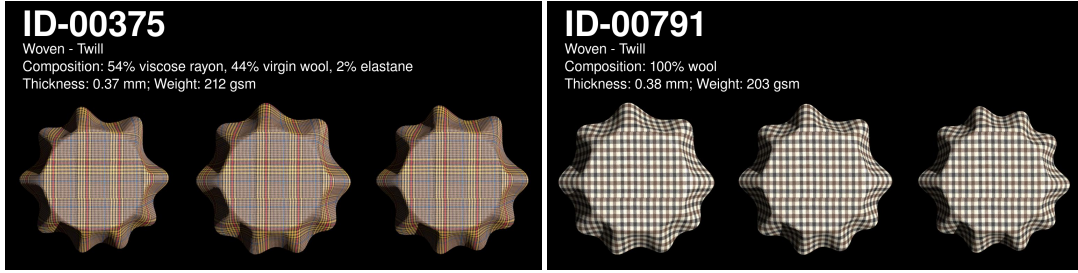


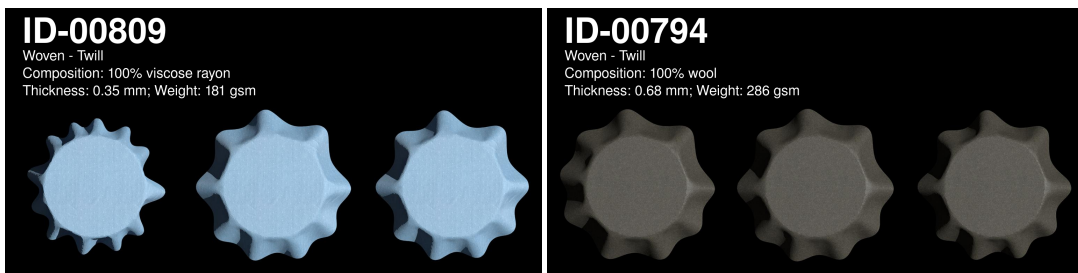
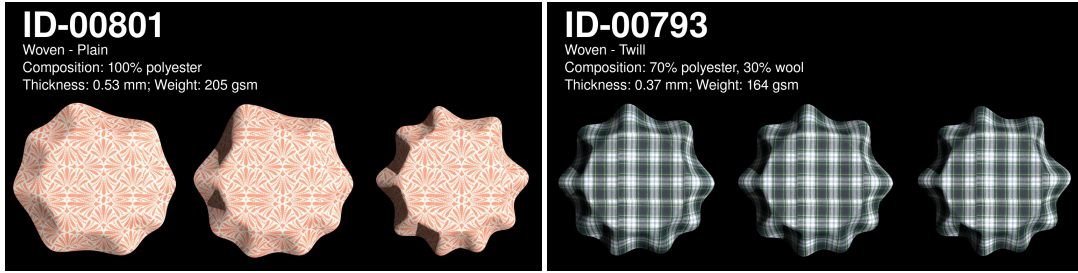
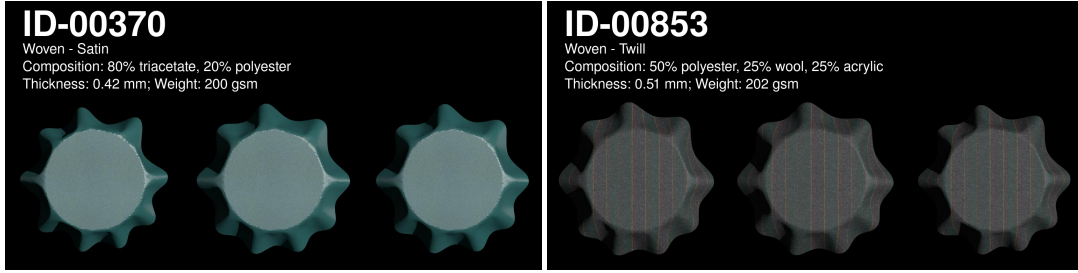


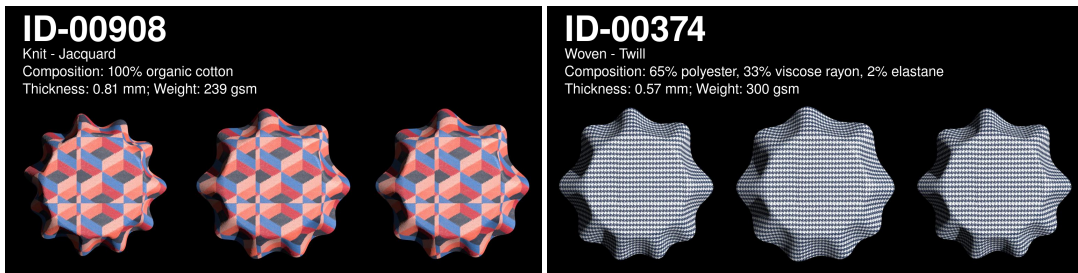
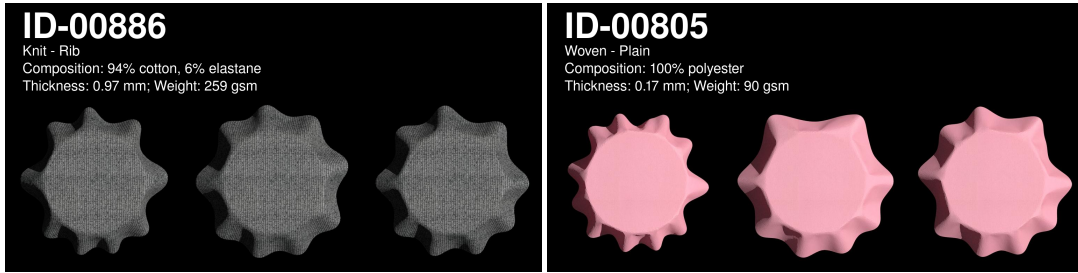












References

- [HZRS15] HE K., ZHANG X., REN S., SUN J.: Deep residual learning for image recognition, 2015. [arXiv:1512.03385](https://arxiv.org/abs/1512.03385). 1