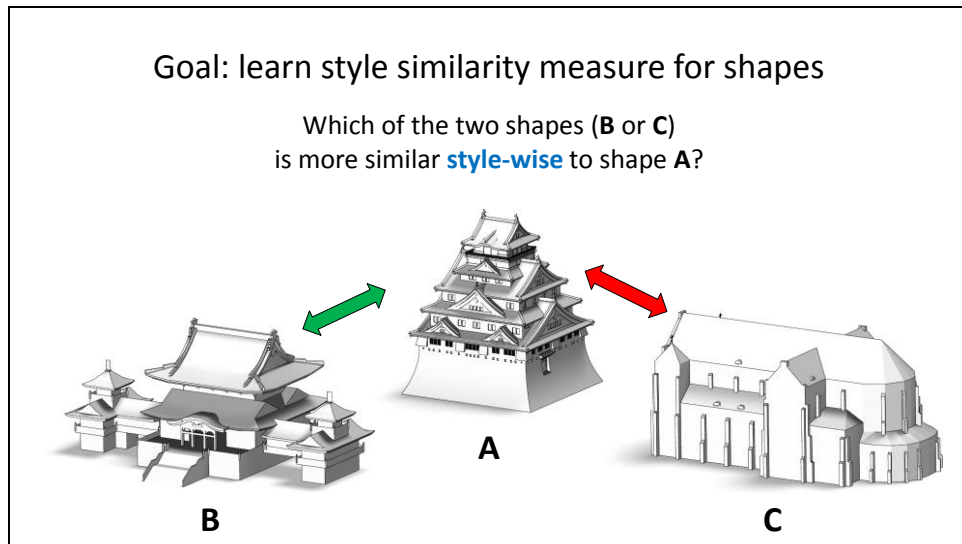
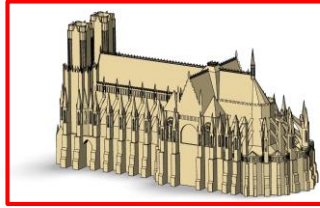


Hi, I am Evangelos Kalogerakis, and together with Zhaoliang Lun, we will present an algorithm for computing a style similarity measure between shapes. This is a joint work between Zhaoliang, me and Alla Sheffer.

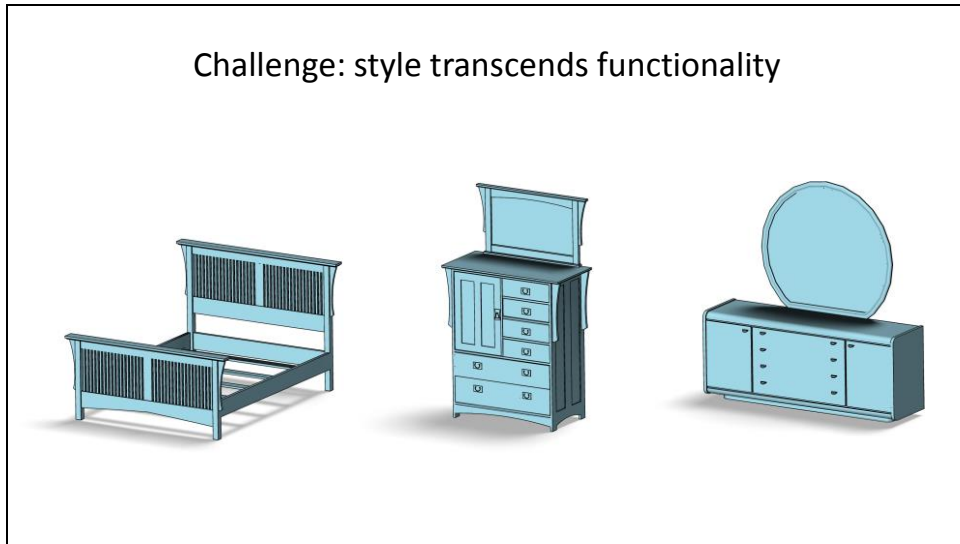


Humans have an innate sense of stylistic similarity. We identify objects as more similar style-wise **[CLICK]**, as in the case of the two Asian temples here, **[CLICK]** or correspondingly we can identify two objects as less stylistically similar. The goal of our algorithm is to compute a measure that replicates this human ability to compare the style similarity of shapes.

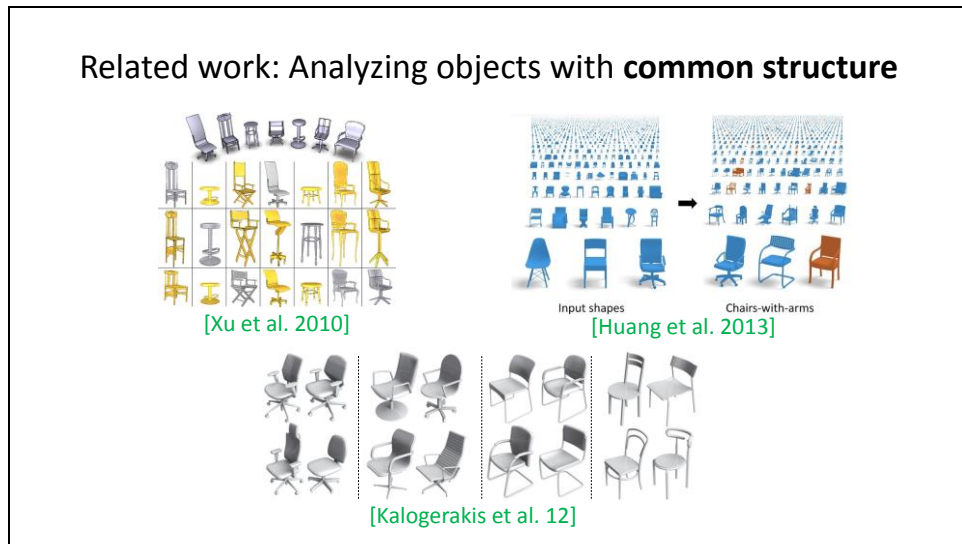
Challenge: style transcends structure



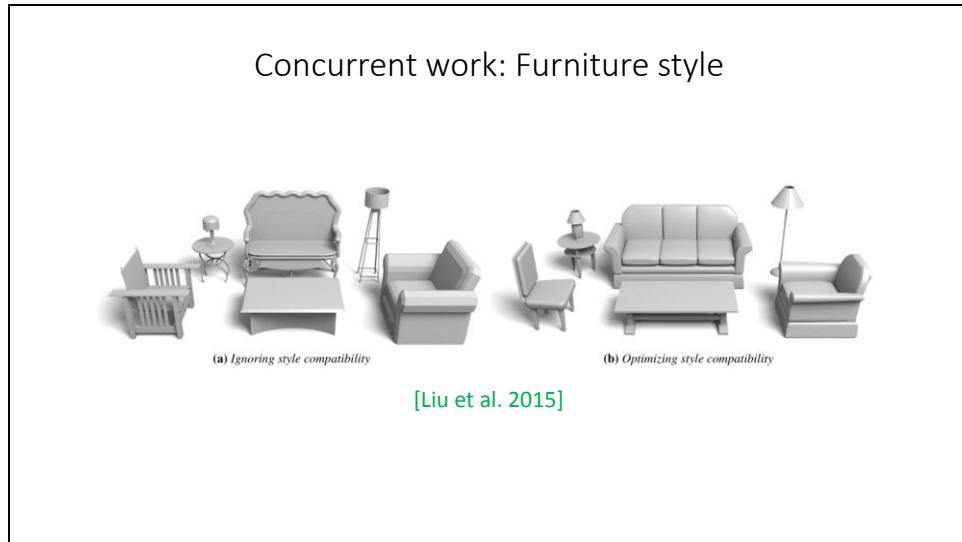
Developing an algorithm that mimics the human perception of style is challenging. Humans intuitively separate style from structure. **[CLICK]** We perceive the Byzantine cathedral and chapel as stylistically similar despite structural differences **[CLICK]** and perceive the Gothic cathedral as very different style-wise.



The perception of stylistic similarity also transcends shape functionality. **[CLICK]** We can pick a bed and a dresser that are stylistically similar to furnish a bedroom, and exclude other stylistically incompatible objects. We want an algorithm to replicate this ability.



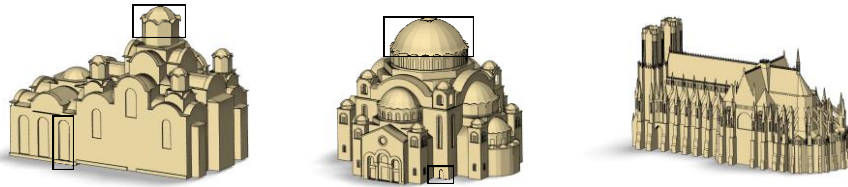
There have been a few prior works that investigated aspects of style for shapes. However, the notion of style that they used was rather coarse and not aligned with human perception. They also could not compare style similarity of shapes with large structural differences, and were mostly limited to cluster or categorize shapes into sub-groups within a particular class e.g., chairs were categorized into rocking chairs, swivel chairs, chairs with long or short legs, and so on. In contrast to these previous methods, our algorithm computes a structure and function transcending style similarity measure, which is well aligned with the human perception of style.



In a concurrent work that will be presented next, Liu et al. introduced a method for measuring style compatibility in the case of furniture. The method assumes input co-segmentations of the furniture shapes – in our case, we do not make this assumption. Our style similarity is also applied to a diverse set of categories, including buildings.

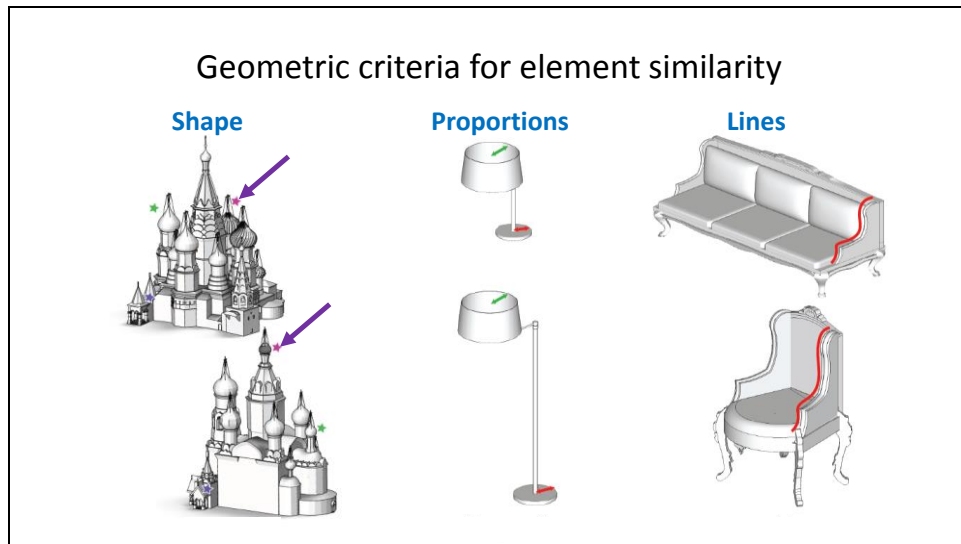
Key observation: Presence of similar style elements

Art literature points to **recurrent similarly shaped, salient geometric elements** as strong indicators of style similarity:



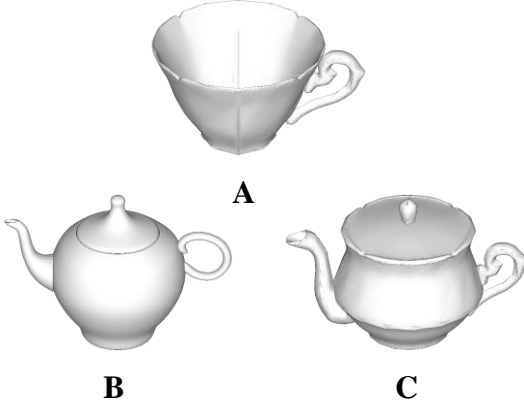
Similar ≠ Identical

In computing our style similarity measure, we are inspired by observations in art history literature. These observations point to the presence of similarly shaped, salient, geometric elements, a key indicator of stylistic similarity. Experts frequently classify objects as belonging to a particular style by detecting approximately similar geometric elements on the objects. **[CLICK]** Examples of such similar geometric elements are the domes, or the doors, you see here in in case of Byzantine churches. **[CLICK]** As you see in this example, the matching geometric elements of shapes do not need to be identical, but only approximately similar.



To measure similarity of style-related elements, the literature points to three important geometric criteria: intrinsic shape **[CLICK]** ... proportions **[CLICK]** ... and lines. The literature also points that style-related elements are frequently designed to be distinctive, or salient **[CLICK]**. Even if we have these valuable observations from art literature in our hands, designing an algorithm for measuring style similarity is still not an easy task. First, we do not know beforehand the number, size, location, area of the style-related elements. Second, even if the literature points to these general geometric criteria for comparing elements, we do not know how to exactly quantify them, or how to translate them into computable geometric features. We also do not know which geometric descriptors developed in the geometry processing literature are the most relevant to these criteria.

Learn measure quantification via crowdsourcing

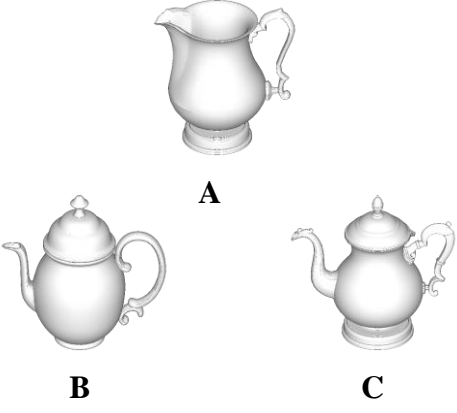


Which of the two objects on the bottom (**B** or **C**) is more similar style-wise to the object on the top (**A**)?

- (i) **B**
- (ii) **C**
- (iii) **Both**
- (iv) **Neither**

We resorted to crowdsourcing and followed a principled approach based on machine learning to quantify these geometric criteria and learn all the parameters in our measure. Crowdsourcing was performed via a large-scale Mechanical Turk study. We ask participants to specify if an object A is more stylistically similar to object B or C displayed with web-based questionnaires.

Learn measure parameters via crowdsourcing



A

B

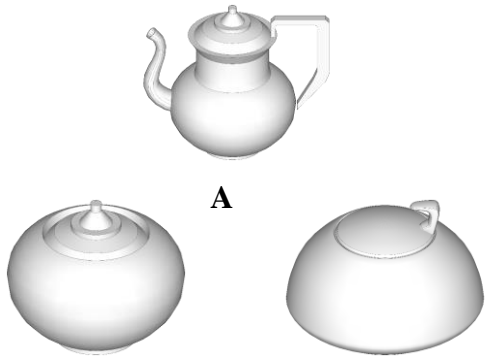
C

Which of the two objects on the bottom (**B** or **C**) is more similar style-wise to the object on the top (**A**)?

- (i) **B**
- (ii) **C**
- (iii) **Both**
- (iv) **Neither**

We released thousands...

Learn measure parameters via crowdsourcing



Which of the two objects on the bottom (**B** or **C**) is more similar style-wise to the object on the top (**A**)?

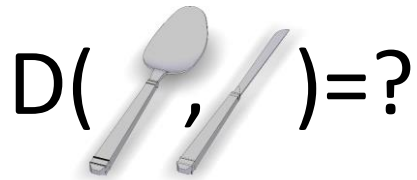
- (i) **B**
- (ii) **C**
- (iii) **Both**
- (iv) **Neither**

...of such questionnaires...

Algorithm for measuring style similarity

Input: a pair of shapes

Output: a measure of style dissimilarity (distance)



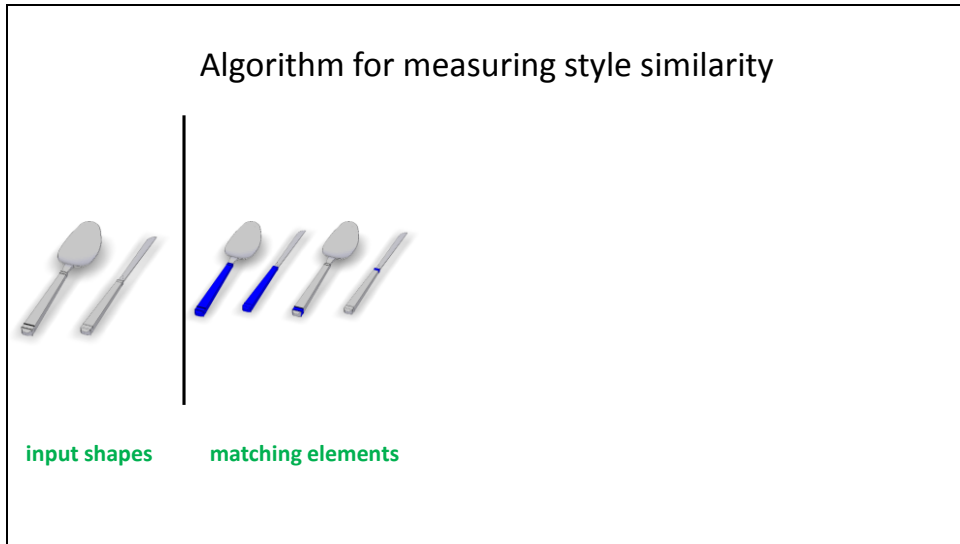
Let's see now how we collated the art history observations, the geometric criteria, and the parameter learning into an algorithmic style measure. Let's start with our problem statement. Given an input pair of shapes, such as **a spoon and a knife here**, **[CLICK]** our goal is to compute a distance measure that quantifies their style dissimilarity.

Algorithm for measuring style similarity

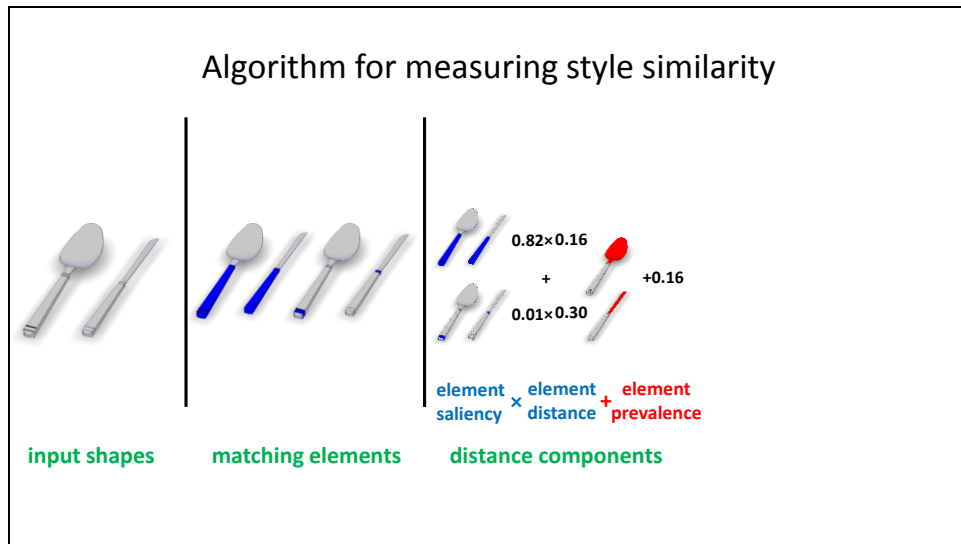


input shapes

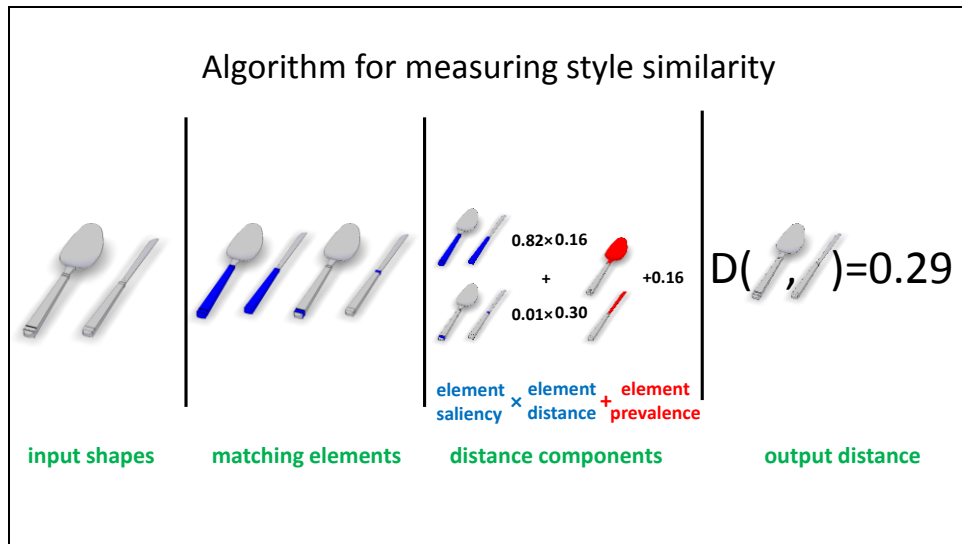
Our algorithm computes the measure with a two-step process. Given the input pair,



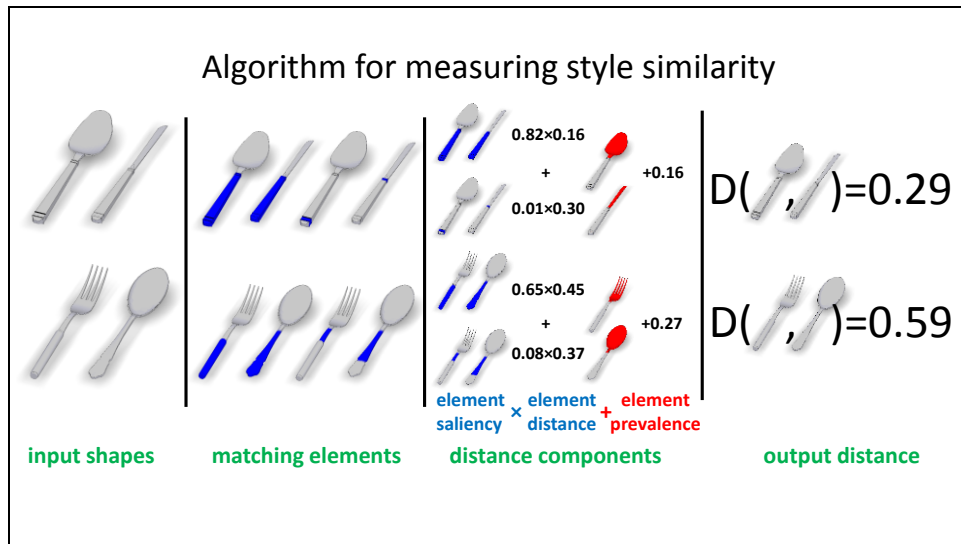
It first identifies matching elements on the input shape pair, such as the handles, or the fine strips, on **the spoon and the knife**



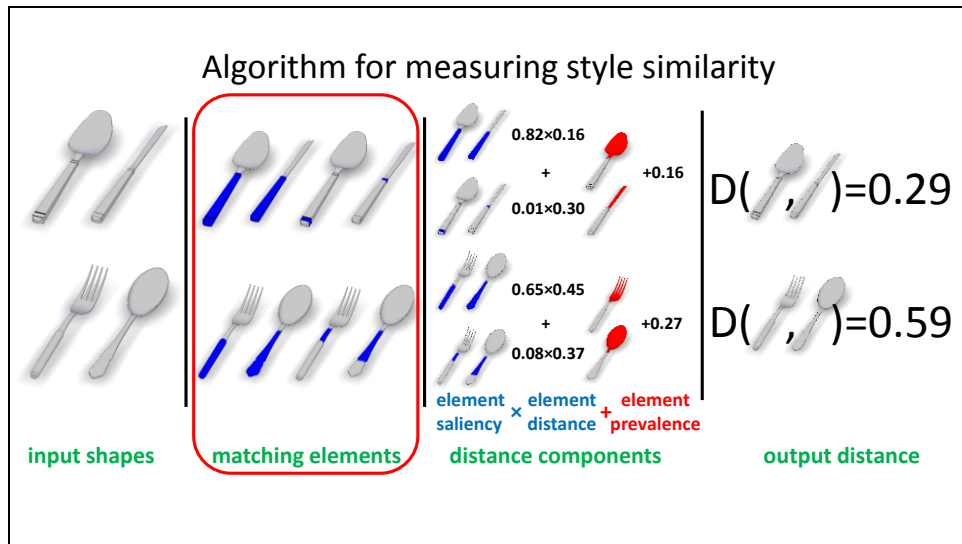
Then it computes the style dissimilarity, or distance, of all pairs of matching element by considering geometric features related to element shape, proportions, and lines. Not all pairs of matching elements are equally important. Thus, their distances are weighted according to element saliency, which is also formulated as a function of geometric features. After summing up the resulting distances between all pairs of elements, we add another term measuring the element prevalence. The term penalizes the area in the two shapes that was not covered by any matching elements, highlighted as red here.



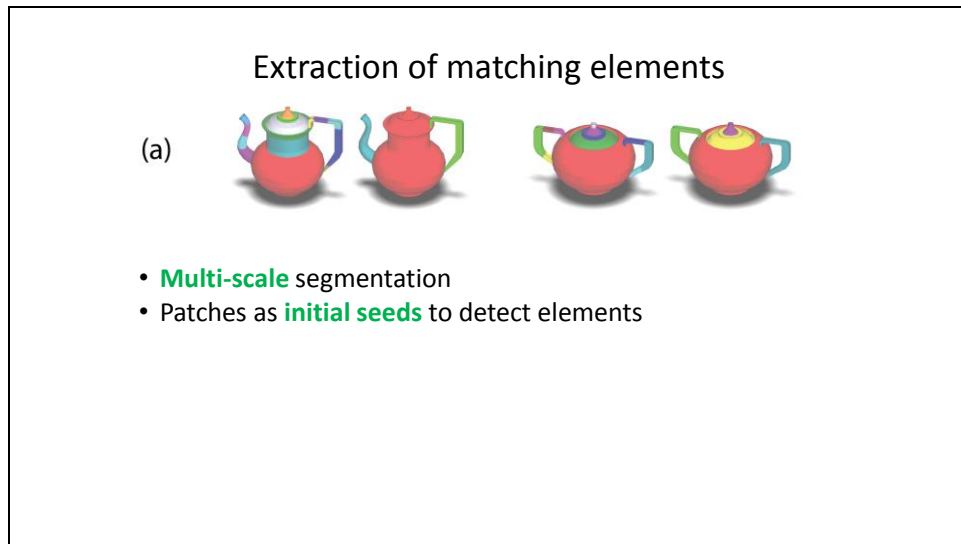
By taking into account the element similarity and prevalence, the algorithm outputs the style dissimilarity measure for these shapes.



The same procedure is followed for any other pair of input shapes. For example, here, based on the computed distances, our method concludes that the **spoon and knife** on the top ... that share similar motifs on their handle ... are significantly more similar style-wise than the **fork and spoon** on the bottom that lack any major similar motifs.





We now describe the element matching step of our method in more detail



Detecting matching elements is challenging since we do not a priori know their size, location, or number, and also the matching elements are not necessarily identical, but only approximately similar. To detect them, we first segment the input shapes into approximately convex patches at multiple scales, as demonstrated for this teapot – we here show two scales of segmentation, **[CLICK]** or similarly for the teapot on the right. **[CLICK]** The patches serve as initial seeds to detect elements.

Extraction of matching elements

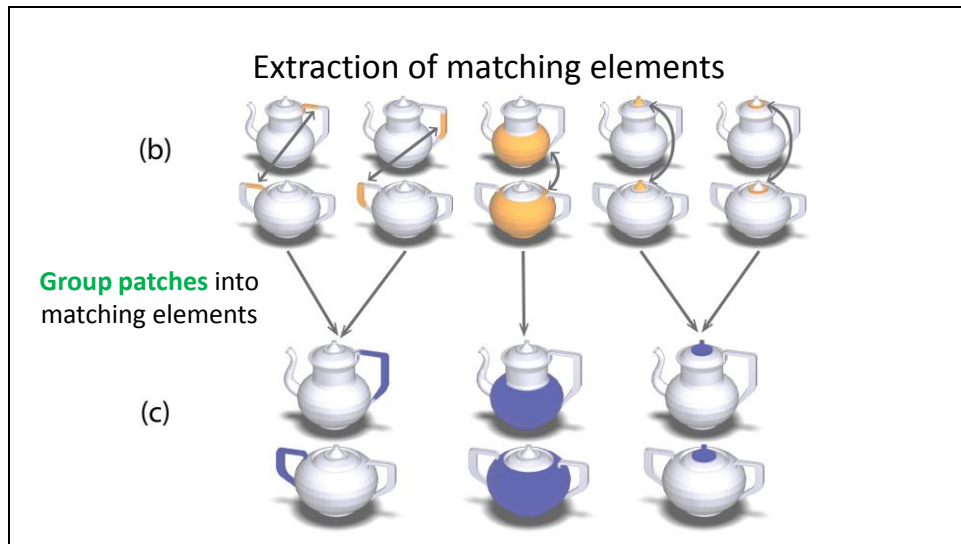
(a) 

(b) 

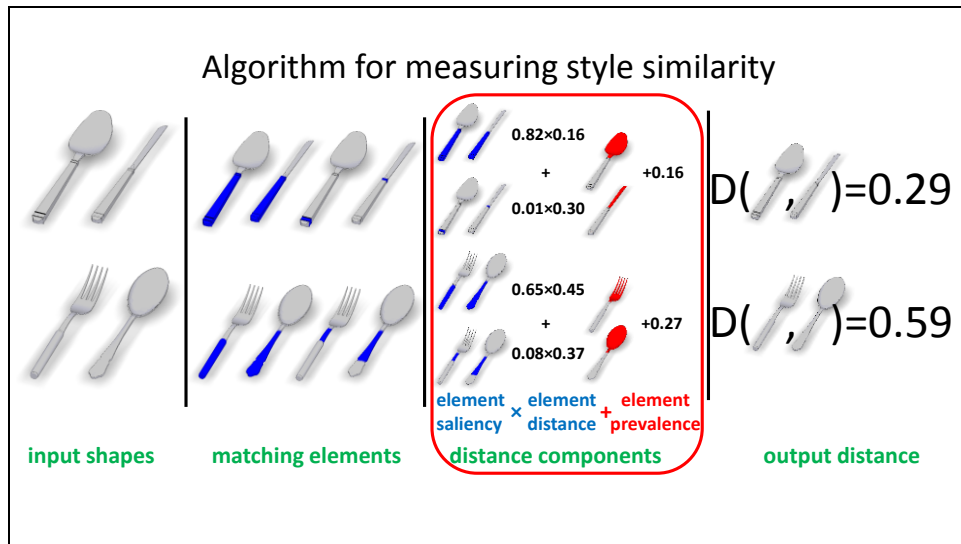
Align with **affine transformation**, measure patch **stylistic similarity**:

$$distance(\text{teapot}_1, \text{teapot}_2) = w_1 \times d_1(\text{handle}_1, \text{handle}_2) + w_2 \times d_2(\text{spout}_1, \text{spout}_2) + w_3 \times d_3(\text{color}_1, \text{color}_2) + \dots$$

Our algorithm then examines the similarity between the pairs of patches on the two shapes. The similarity is examined by approximately aligning them with an affine transformation, **[CLICK]** then evaluating a distance measure between patches expressed as a weighted combination of elementary distances on their individual geometric features, such as surface point-to-point distance, feature curves distances, curvature histograms distances. The weights of this distance measure are learned from the crowdsourced data.



Contiguous pairs of patches with high similarity are grouped into matched elements **[CLICK X 3]**. Grouping is formulated as a min-cut labeling problem. We refer to the paper for more details.



The detected pairs of matched elements are used as input to the second step of our method that computes the components of our style measure between the two shapes. For each pair of detected elements, we measure their distance **[HIGHLIGHT WITH LASER]**

Element distance

Weighted combination of distances on **individual geometric features:**

$$\text{distance}(\text{fork}, \text{spoon}) = w_1 \times d_1(\text{handle}, \text{handle}) + w_2 \times d_2(\text{head}, \text{head}) + w_3 \times d_3(\text{tip}, \text{tip}) + \dots$$

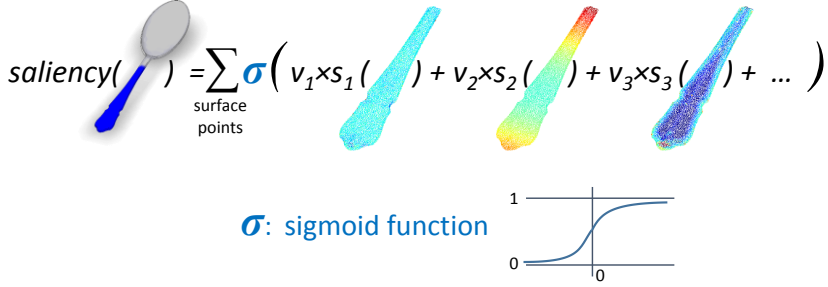
The distance between elements is evaluated with the same weighted combination of elementary distances used in the element matching step, such as surface point-to-point distance, curvature histograms distances and so on.

Element saliency

Weighted combination of **individual geometric features**:

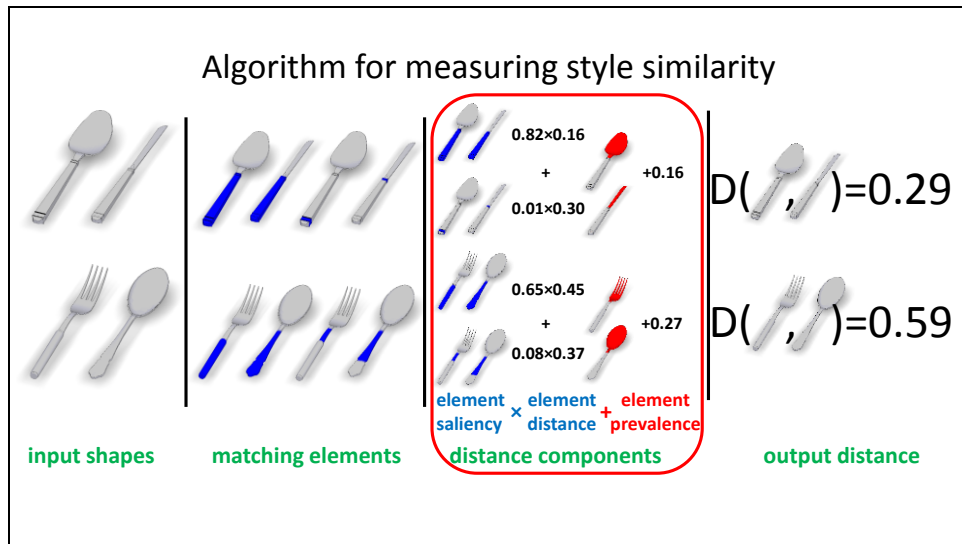
$$\text{saliency}(\text{spoon}) = \sum_{\text{surface points}} \sigma \left(v_1 \times s_1(\text{spoon}) + v_2 \times s_2(\text{spoon}) + v_3 \times s_3(\text{spoon}) + \dots \right)$$

σ : sigmoid function



The diagram illustrates the calculation of element saliency for a spoon. It shows the spoon and three heatmaps representing different geometric features (s1, s2, s3) measured over surface points. A sigmoid function graph is also shown, indicating that the resulting weighted combination is non-linearly scaled.

The distance between elements is weighted by their saliency. The saliency of each element is computed as a weighted combination of geometric features, such as surface curvature, ambient occlusion and so on, measured over the surface points. The resulting weighted combination is non-linearly scaled through a sigmoid function.



Finally, we add the element prevalence term to our measure [**HIGHLIGHT WITH LASER**]

Element prevalence

$$D_{prevalence}(\text{fork}, \text{spoon}) = \frac{\text{saliency}(\text{fork}) + \text{saliency}(\text{spoon})}{2} \cdot t$$

t : learned penalty parameter

This term measures the percentage of the area on both models not covered by any matched elements, weighted by their saliency. The unmatched area is penalized with a penalty parameter t .

Parameter learning

Learn parameters from training triplets:

- element-similarity weights (\mathbf{w})
- saliency weights (\mathbf{v})
- prevalence penalty (t)

All the parameters of our style measure, including **[CLICK]** the weights used in the element similarity term, **[CLICK]** the weights of the element saliency term, **[CLICK]** and the penalty parameter of the prevalence term, are learned from the crowdsourced training data.

Parameter learning

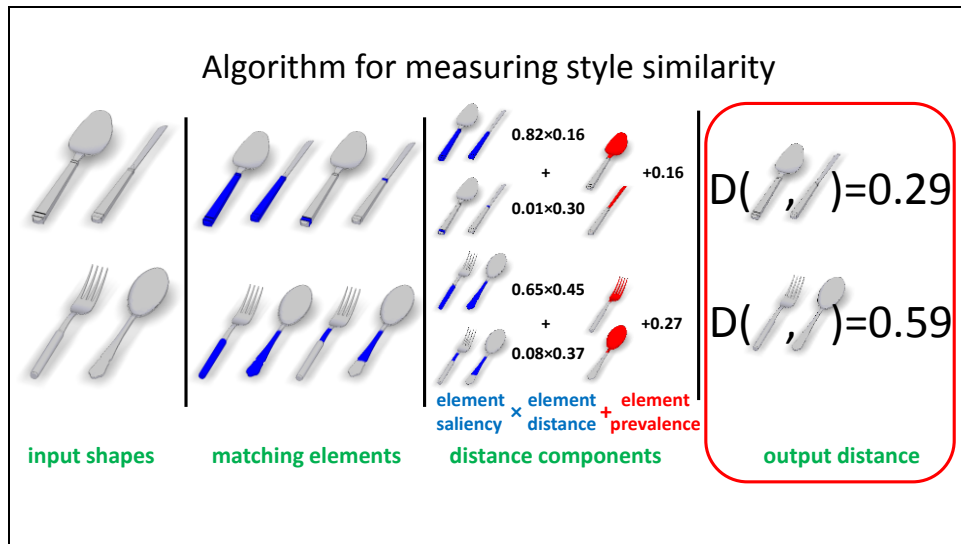
Learn parameters from training triplets:

- element-similarity weights (\mathbf{w})
- saliency weights (\mathbf{v})
- prevalence penalty (t)

that maximize likelihood function & regularizer to promote sparsity:

$$L(\mathbf{w}, \mathbf{v}, t) = \sum_{\text{triplet } \{A,B,C\}} \text{confidence}(B) \cdot \log P(\text{B is more similar to A than C}) \\ + \sum_{\text{triplet } \{A,B,C\}} \text{confidence}(C) \cdot \log P(\text{C is more similar to A than B}) \\ + \text{regularizer}(\mathbf{w}, \mathbf{v}, t)$$

The parameters are learned such that the agreement between our measure and participant responses is maximized. To do this, we form a likelihood function for our parameters set that measures the degree of agreement between the crowdsourced relative comparisons of style and our style measure under all different parameter values. Our goal is to maximize this likelihood function together with a regularization term **[CLICK]**, which promotes sparsity in the learned weights. We regularize the L1 norm of the weights to help us discover the most relevant geometric features for style. For more details, we refer to our paper.



We now have all the components for our style similarity measure together with their learned parameters, and can compute the style dissimilarity between any pairs of shapes.

Validation



Does it work?

Let's discuss now the validation of our method.

Our dataset

>50,000 responses

More than **2,500 participants**

Seven diverse categories with **1,000 shapes:**

buildings, furniture, lamps, columns, cutlery, coffee set, dishes

First, a few words about the dataset we gathered from Mechanical Turk and used for training and validating our method. We gathered over 50,000 such style comparisons **[CLICK]** from more than 2,500 Mechanical Turk participants **[CLICK]** in seven diverse shape categories including about 1,000 shapes, such as buildings, furniture, cutlery and so on.

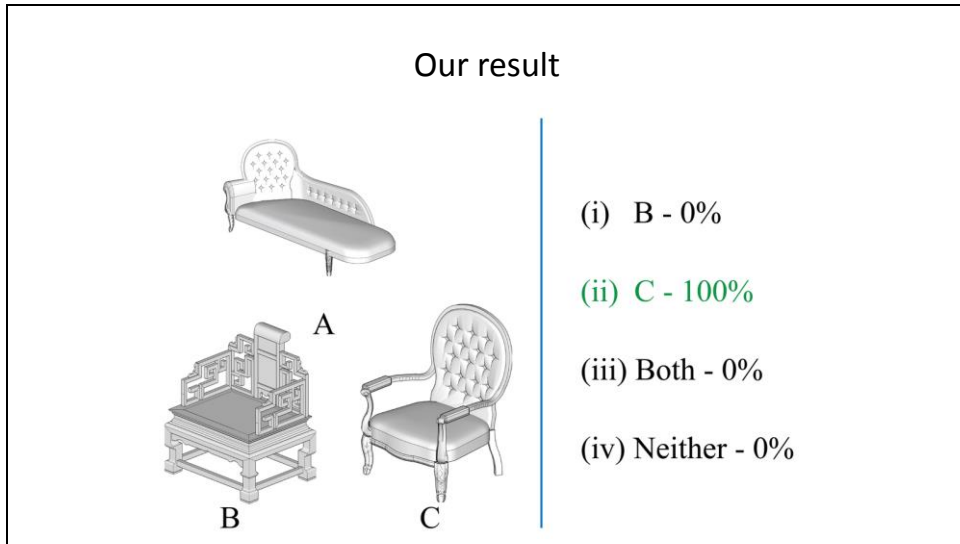
Are users consistent in performing relative style comparisons?

Category	% consistency
building	76.8%
furniture	86.0%
lamp	89.8%
column	87.3%
coffee set	83.5%
cutlery	89.4%
dish	78.7%
total	85.0%

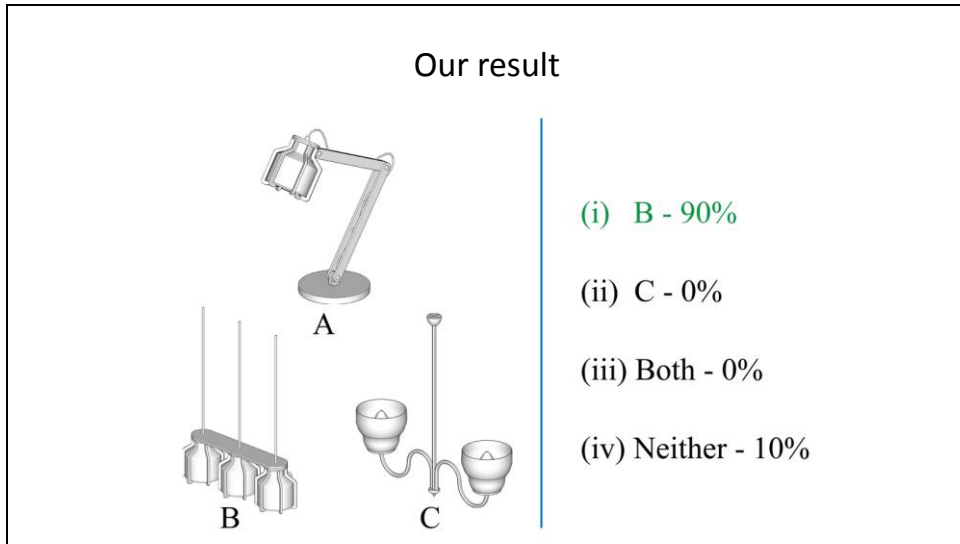
Well, the first question to ask is whether it makes sense to have a style similarity measure in general. If the participants were NOT consistent in picking responses to the given style similarity comparisons we provided to them - in other words, if they did not largely give the same answer to our queries, then obviously it would not make any sense to perform any form of evaluation, train our algorithm, or develop a style similarity measure in general. This was not the case: on average **[CLICK]** 8.5 out of 10 users agreed with the plurality answer per query, confirming our hypothesis that users are consistent in performing relative comparisons of style similarity. Does our learned style similarity measure also agree with the participants' plurality response?

Our result		
Category	% consistency	our measure
building	76.8%	81.4%
furniture	86.0%	91.4%
lamp	89.8%	95.0%
column	87.3%	90.2%
coffee set	83.5%	90.6%
cutlery	89.4%	85.8%
dish	78.7%	89.5%
total	85.0%	89.1%

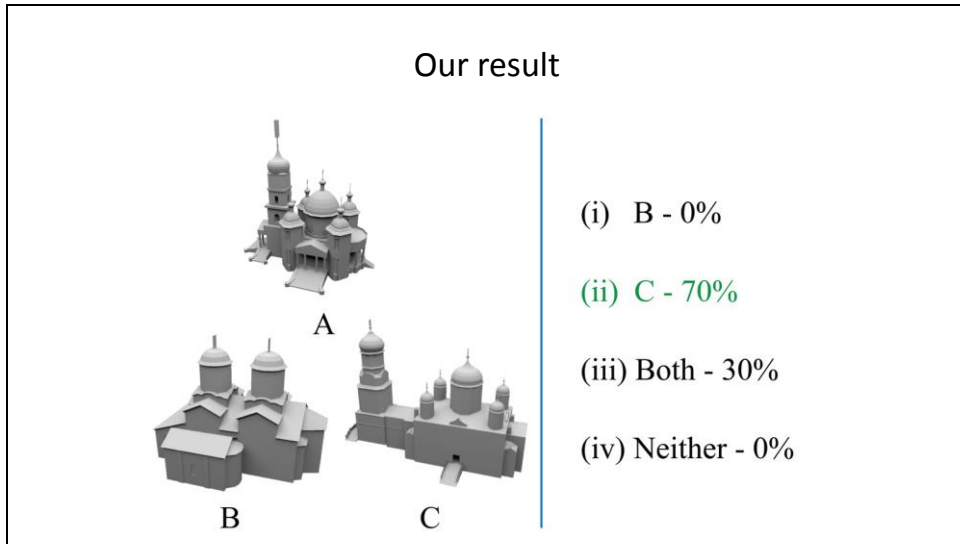
The answer is yes! Our style measure was validated against participant responses using ten-fold cross-validation. We found that **[CLICK]** 89% of the time on average our measure agrees with the plurality response in the queries.



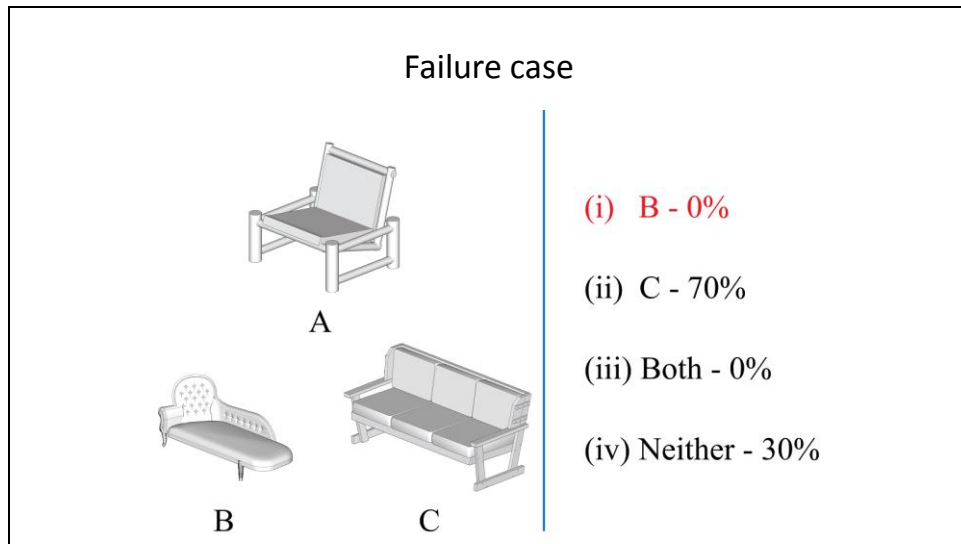
Here are characteristic examples of queries where our measure agrees with the users' plurality response. For example, in this query, 100% of the users indicated that C is more similar style-wise to A compared to B. Our measure also gave the same response.



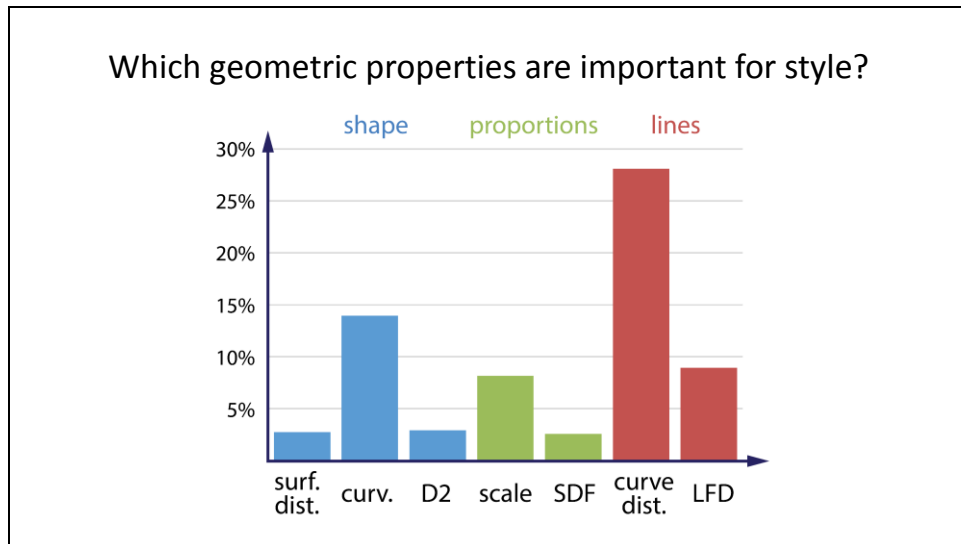
Here is another query. Our algorithm gives the correct response despite all the structural differences in the lamps.



Our algorithm also gives correct answers even for structurally complicated shapes, such as buildings.



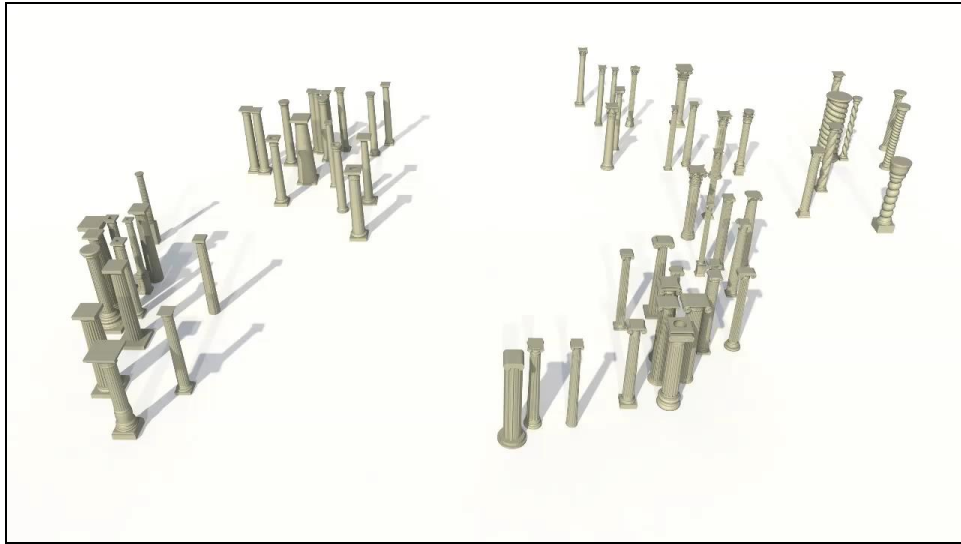
Yet, our algorithm also has failure cases due to potential element mismatches.



Our learned measure can lead to interesting remarks regarding the importance of the different geometric features used for comparing the style similarity of shapes. We analyzed the relative importance of each geometric feature reflected by our learned weights in the element-similarity terms averaged over our seven categories. We found that **[HIGHLIGHT]** comparing the similarity of feature curves, including ridges, valleys, and silhouettes play the most dominant role in our similarity measure, followed by the element surface curvature and element scale. These remarks agree with the art literature observations, which we discussed earlier in our observations, pointing to the use of lines, intrinsic shape, and proportions to compare elements.

Application:
Organizing shape collections

We now describe applications of our style similarity measure. First our measure can be used to organize shape collections such that users visually explore groups of shapes based on their style.



We demonstrate this application in the case of columns. The produced pairwise distances of our style measure were used to embed our dataset of columns into a two-dimensional space. Our embedding yielded distinct clusters that corresponded to well-known architectural orders

[, such as Doric ... Roman Doric ... Corinthian ... Ionic ... and Spiral-style columns]

Application:
Style-based shape tagging

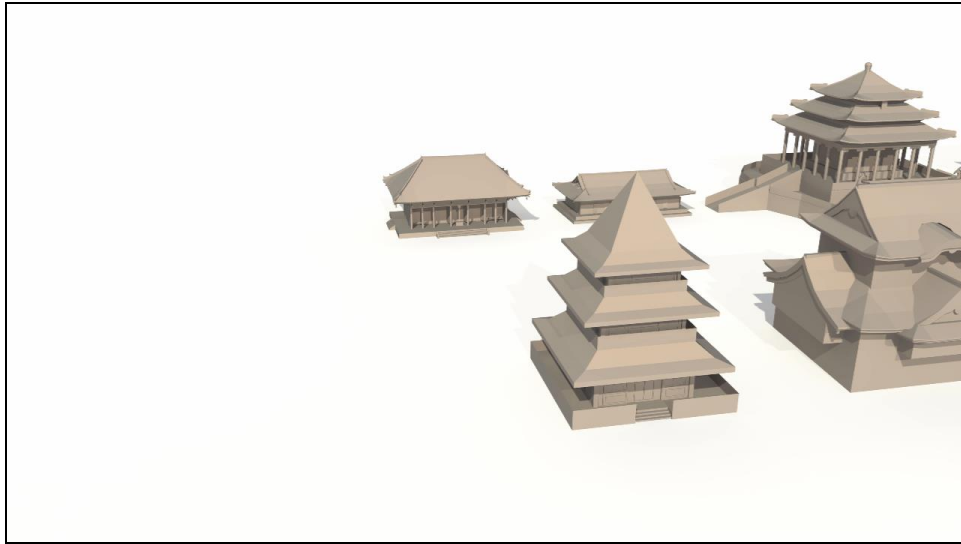
We can also use our distance function to infer style-related tags to enable keyword based search.

Slide 42



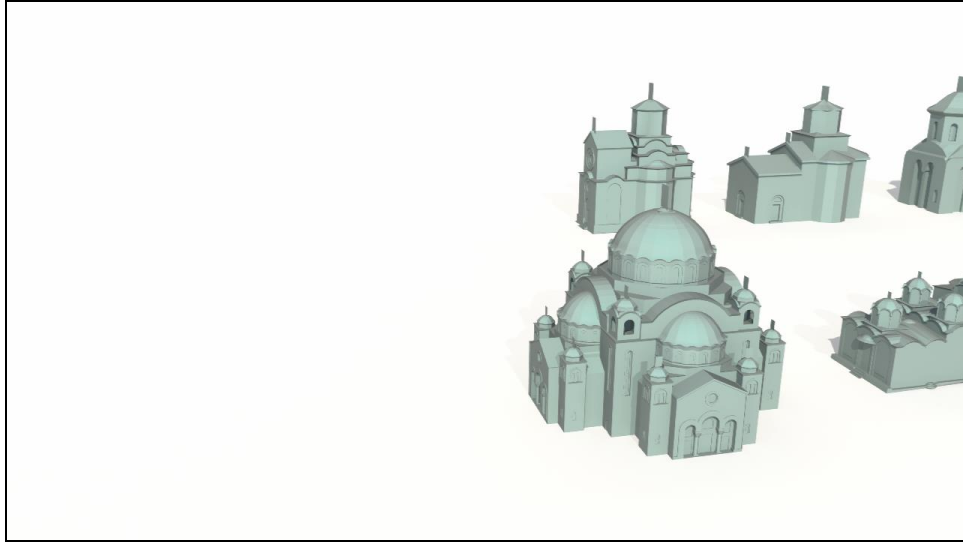
Given a small set of training shapes with style labels provided by an expert, such as the ones shown here for buildings ... our algorithm can propagate these labels to other shapes.

Slide 43



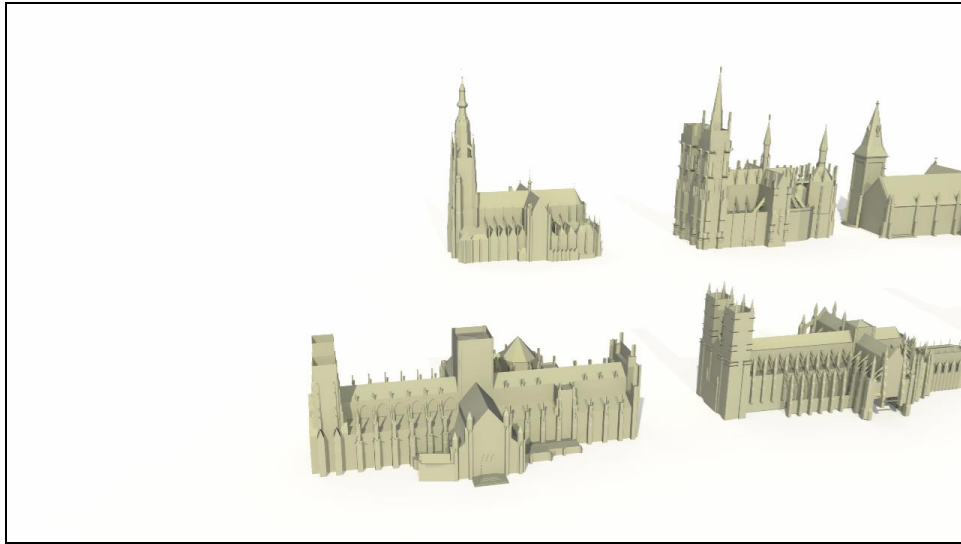
Here we show buildings that our algorithm successfully labeled as Asian temples...

Slide 44



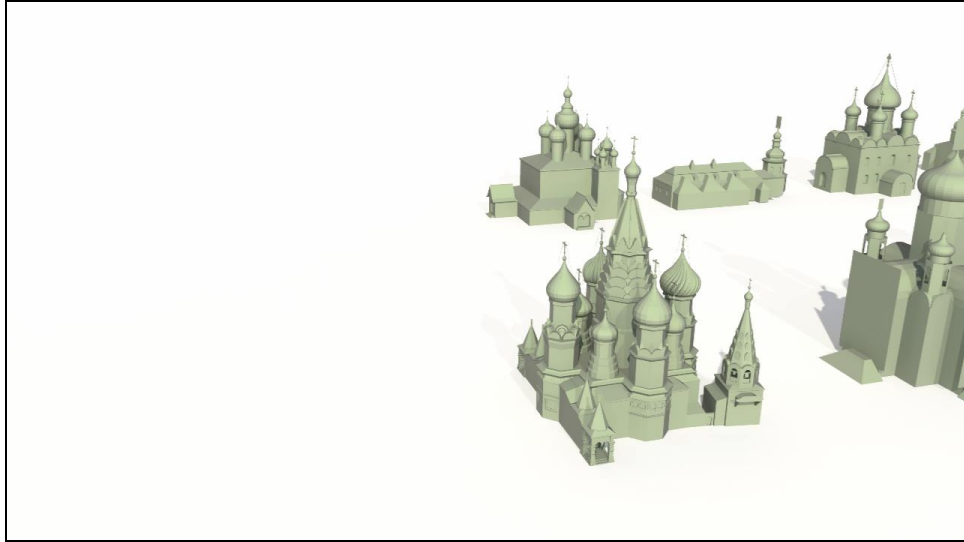
...Byzantine churches...

Slide 45



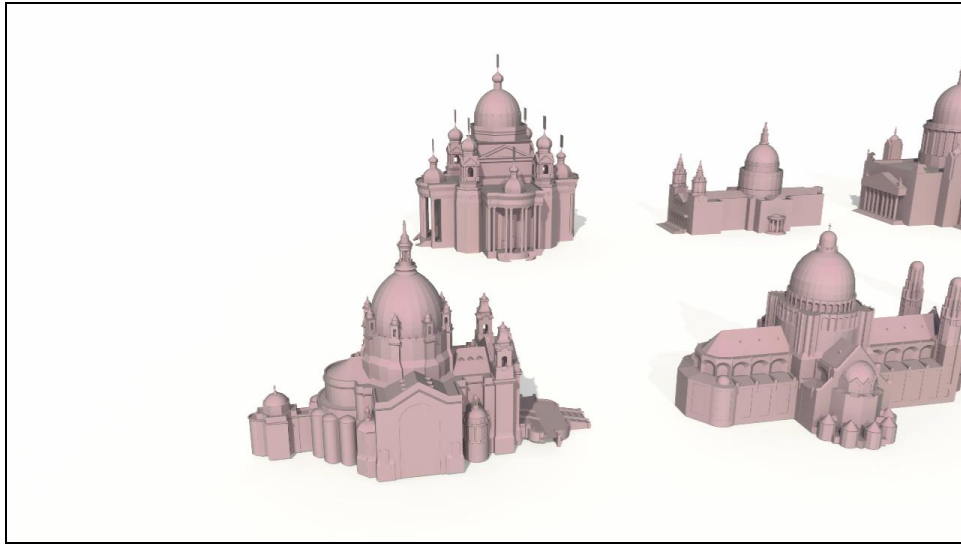
...Gothic buildings...

Slide 46



...Russian churches...

Slide 47



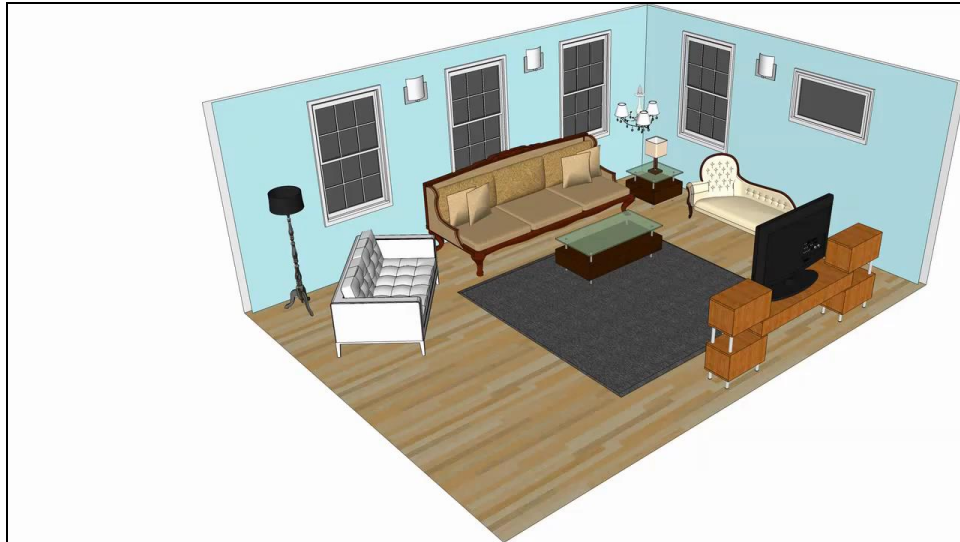
...and finally Baroque-era buildings.

Application:

Style-based suggestions for scene modeling

We can also use our measure to provide style-based suggestions for scene modeling.

Slide 49



For example, here we show a scene where furniture items have incompatible style. The user specifies a query furniture item ... and our application retrieves pieces of furniture with similar style ... Here the user specifies a query lamp ... and our application also retrieves stylistically compatible lamps. [**SYNCHRONIZE WITH VIDEO**]

Summary

- First algorithm for computing a structure transcending style similarity measure between shapes
- Key components: geometric style elements + crowdsourcing-based learning
- Demonstrated to be well aligned with human perception of style

To summarize, we presented the first algorithm to compute a structure-transcending style similarity measure between shapes. The key components of our algorithm is the detection of geometric style elements and the parameter learning to quantify the geometric criteria used in our similarity measure based on crowdsourced data. Our algorithm is demonstrated to be well aligned with human perception of style based on our experiments.

Thank you!

Acknowledgements: *Nicholas Vining, Mikhail Bessmeltsev*

Our project web page:

<http://people.cs.umass.edu/~zlun/papers/StyleSimilarity>



Thank you for your attention! We show here our project page that includes source code, our dataset, and results.