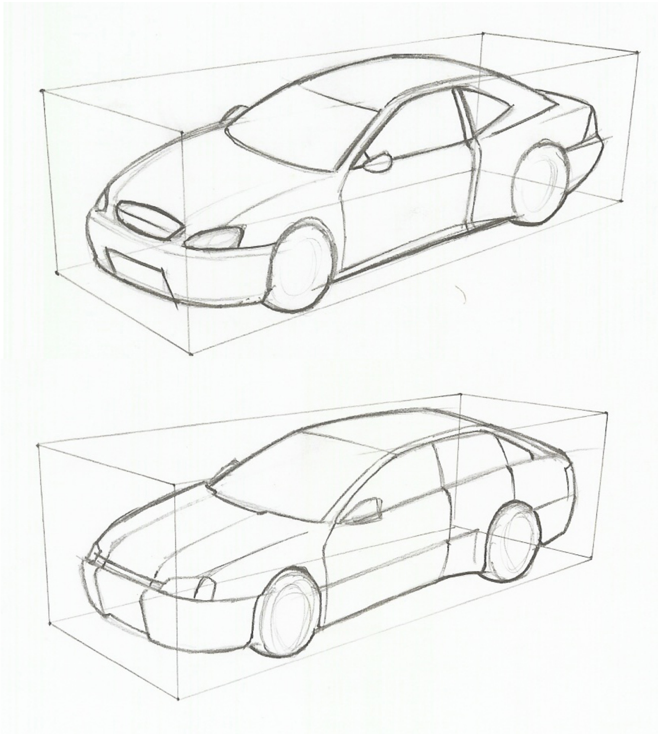
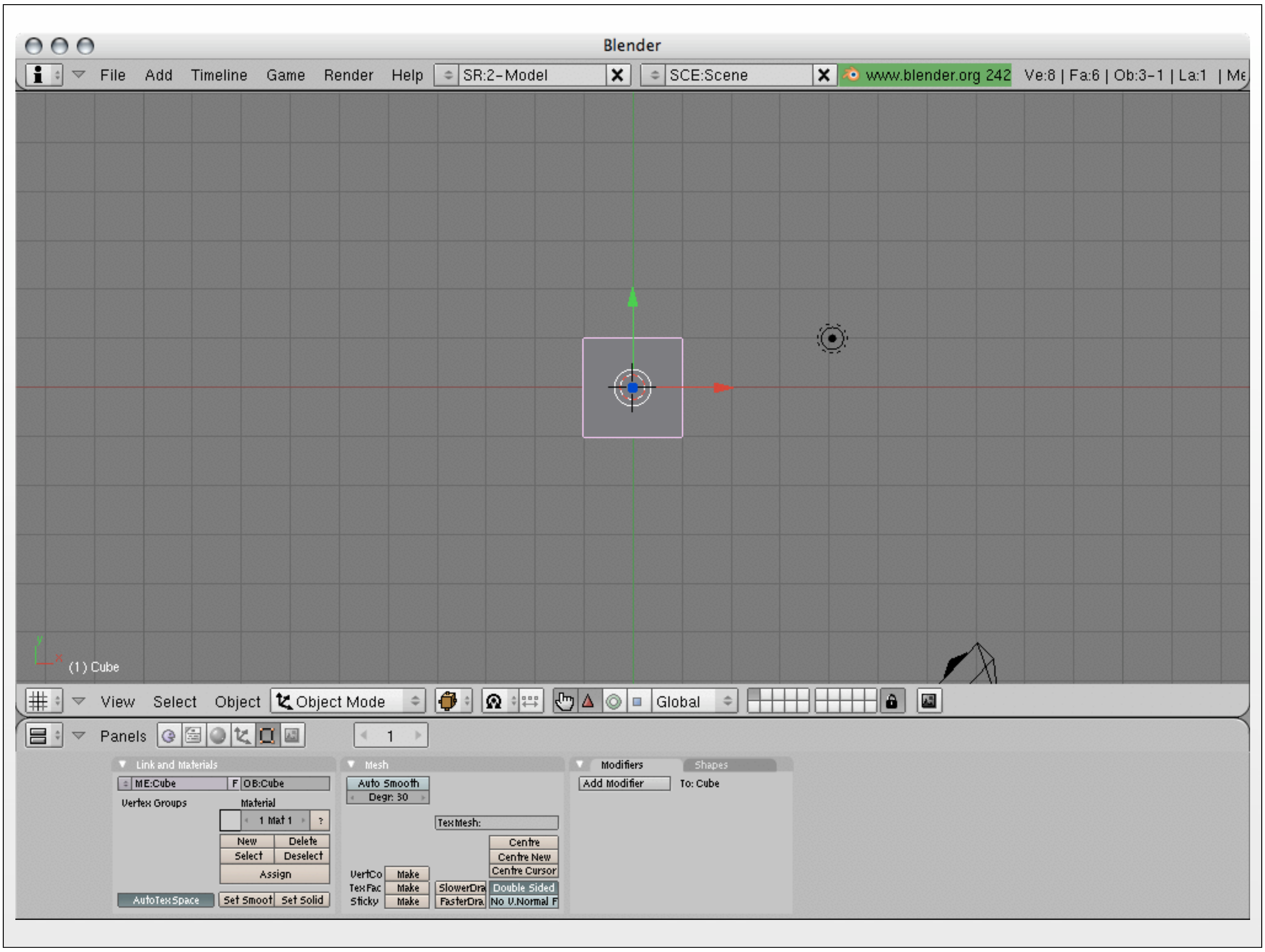
Example-Based Conceptual Styling Framework for Automotive Shapes

I. Kókai, J. Finger, R.C. Smith, R. Pawlicki, T. Vetter
University of Basel + General Motors R&D

Hello Everyone, Today I will be talking about a framework we developed for conceptual automotive design or styling. In our work we concentrated on the early phases of design or styling where traditional 2d methods,...



... like sketching are still pretty much the norm. Design is an explorative process, where the ideas emerge through the process, where the very act of doing/creation gives birth to new ideas. It is crucial for the tools supporting this process that they don’t get in the way. Unfortunately many 3d modeling packages are tailored more for engineering than for the designers. The designers are confronted with interface like this...

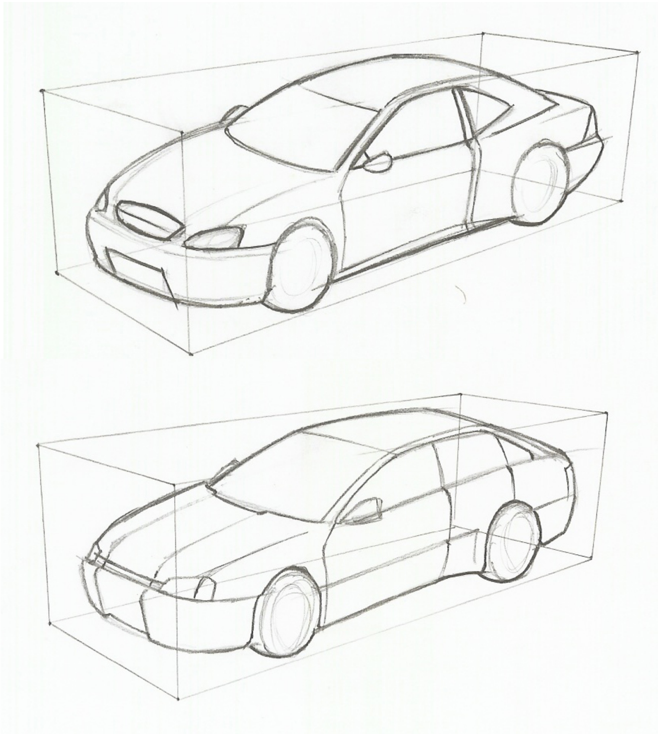


... Where they have to think about different modes, they have to think about the shape as a sum of parameteric patches and curves, and control points, etc. And as Stephen Weston put it in an article about computer aided industrial design, they cry out loud...

“Dammit Jim, I’m a designer, not a mathematician”

Stephen H. Westin: Computer Aided Industrial Design
Computer Graphics, February 1998

Dammit Jim, I am a designer not a mathematician. And they go back to what they know...



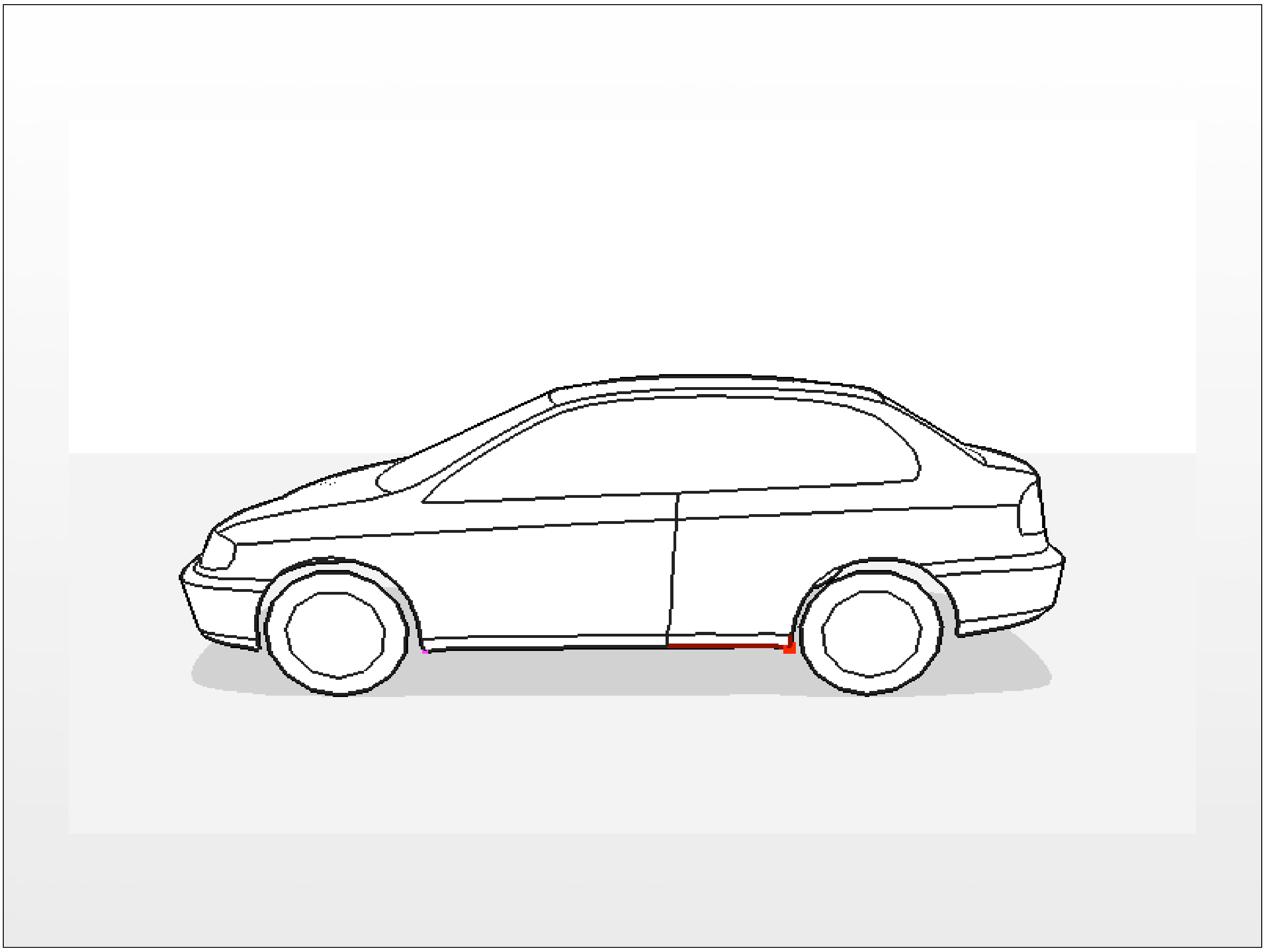
5

... pen and paper or a tablet with a 2d sketching software. The problem is with this, that the 2d drawings are ambiguous for anyone apart from the designer who draw it. To really evaluate the surface we need 3d representations, which have to be created from the chosen sketches first. This is most of the time a lengthy process...

6

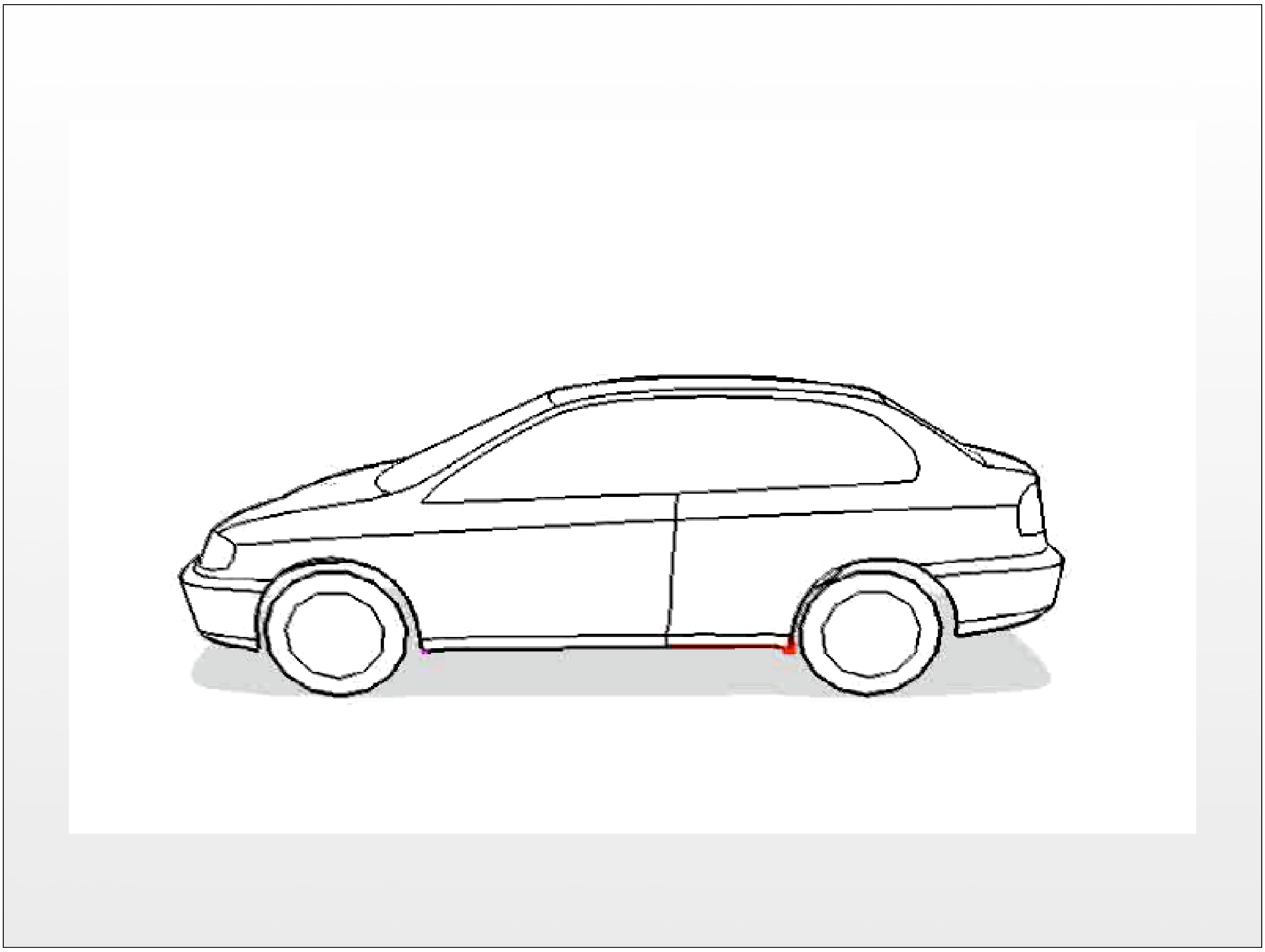
... Of course it is not looking like this anymore. But still a modeler is needed who converts the sketches into CAD models which can then be evaluated on the screen, or virtual environments or milled in clay. But this is a lengthy process and if modifications are needed than we should go back to the drawing table. To eliminate this bottleneck in the process, we would need to give a tool for the designer where he can draw or model a new shape in 2d using his traditional skill set, but still have any time during the process a 3d model...

In the next couple of slides I'll walk through our proposed workflow with a couple of videos demonstrating our system...

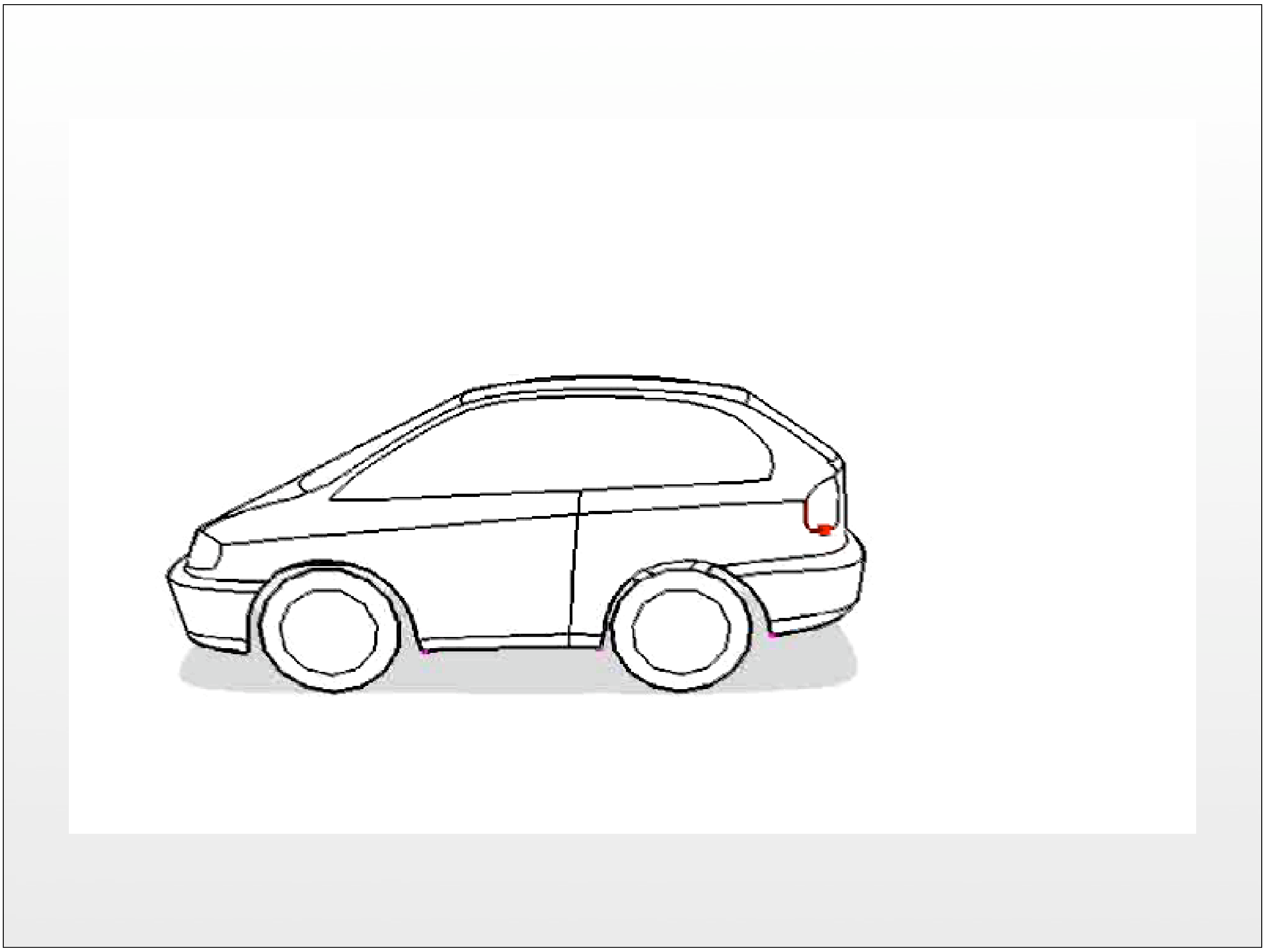


8

In our system the designer starts with a 3d model rendered on the screen, which can be some earlier design, a shape generated from a 2d sketch, or something else...

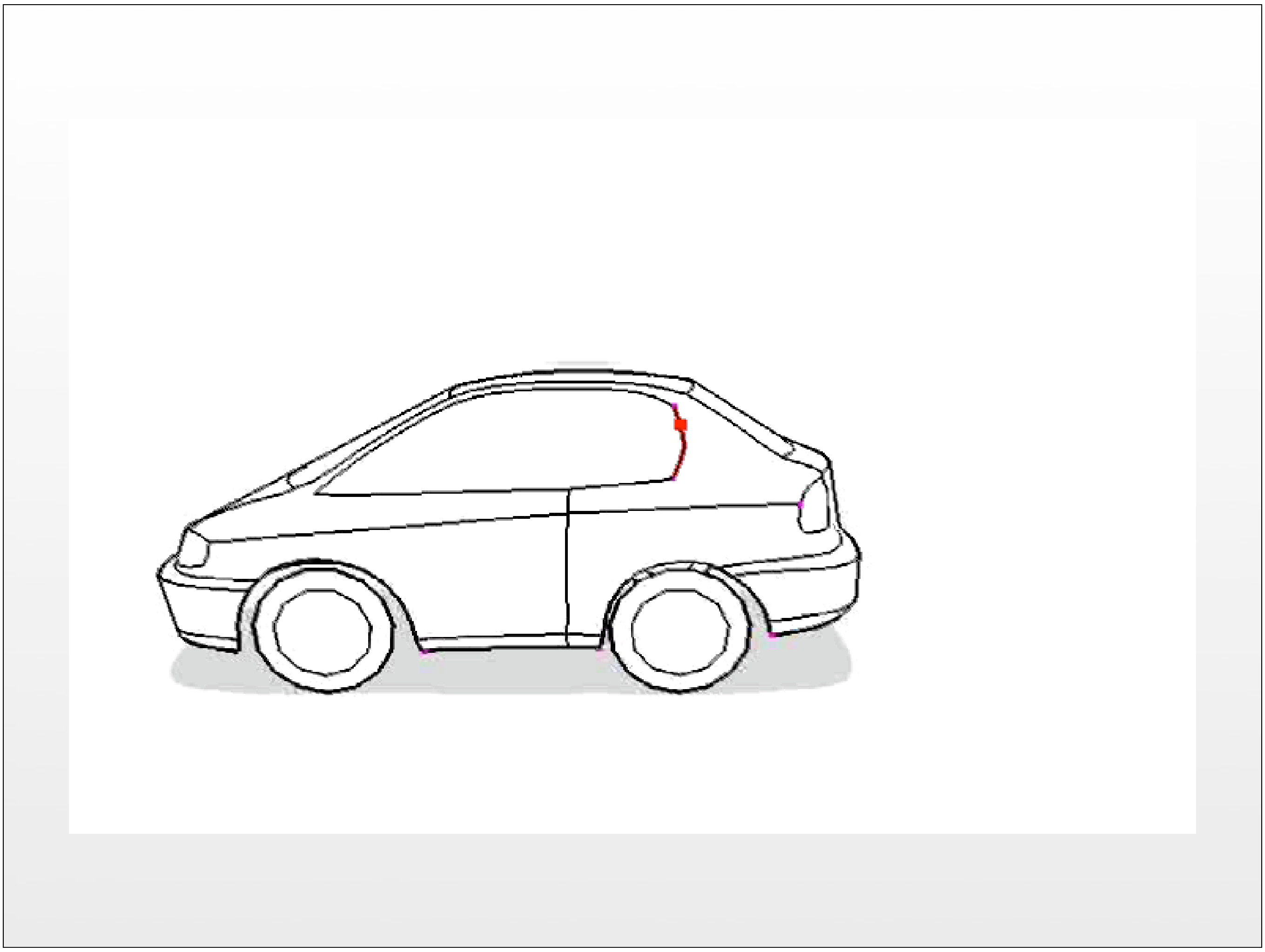


... he starts to modify it with dragging points of the shape and pulling them to change the basic proportions of the shape, notice the explorative nature of the process...



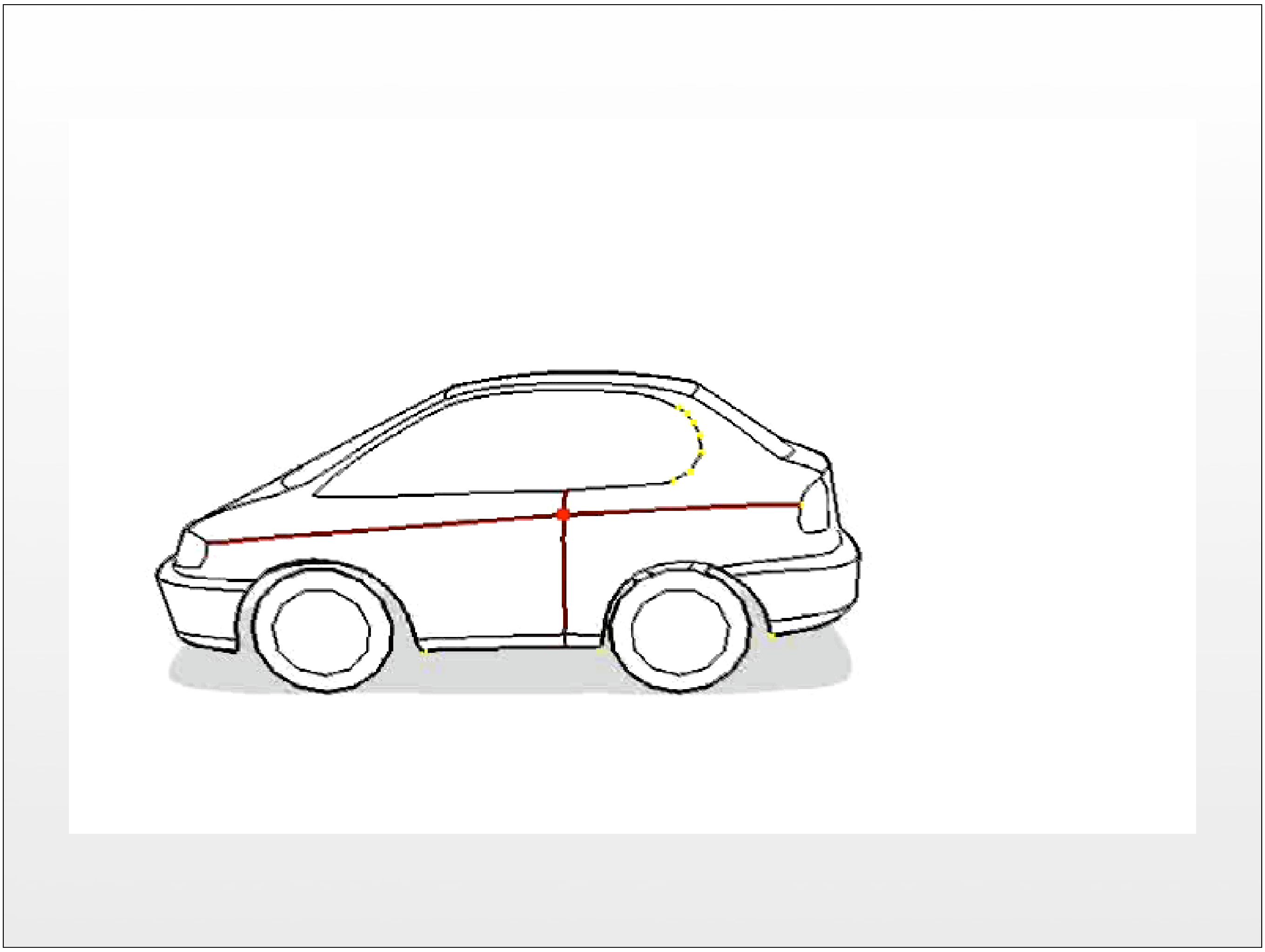
10

He can do this for several points, and get a rough shape...
Anytime during the process...



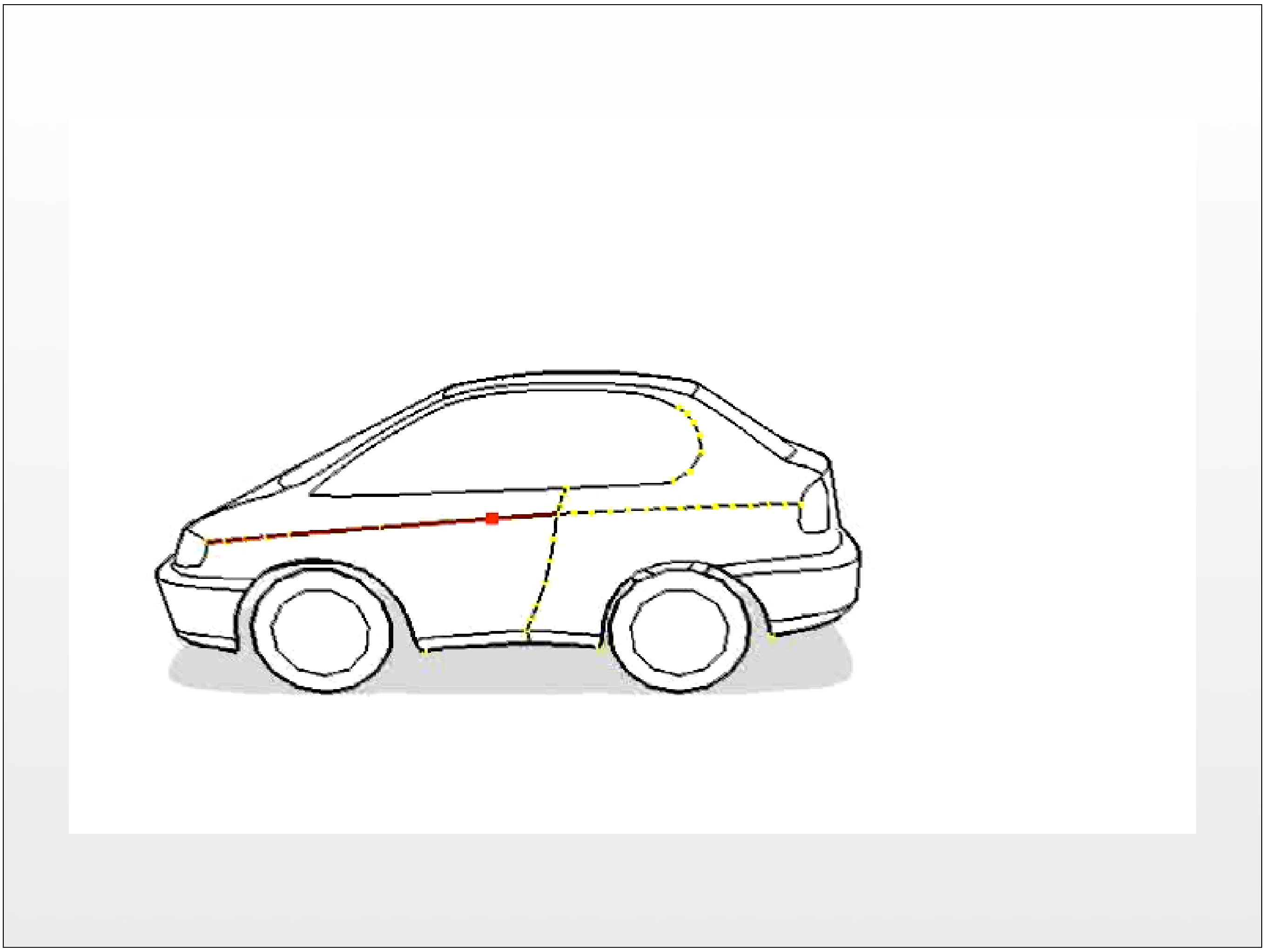
11

He can draw over the shape to modify it more precisely...

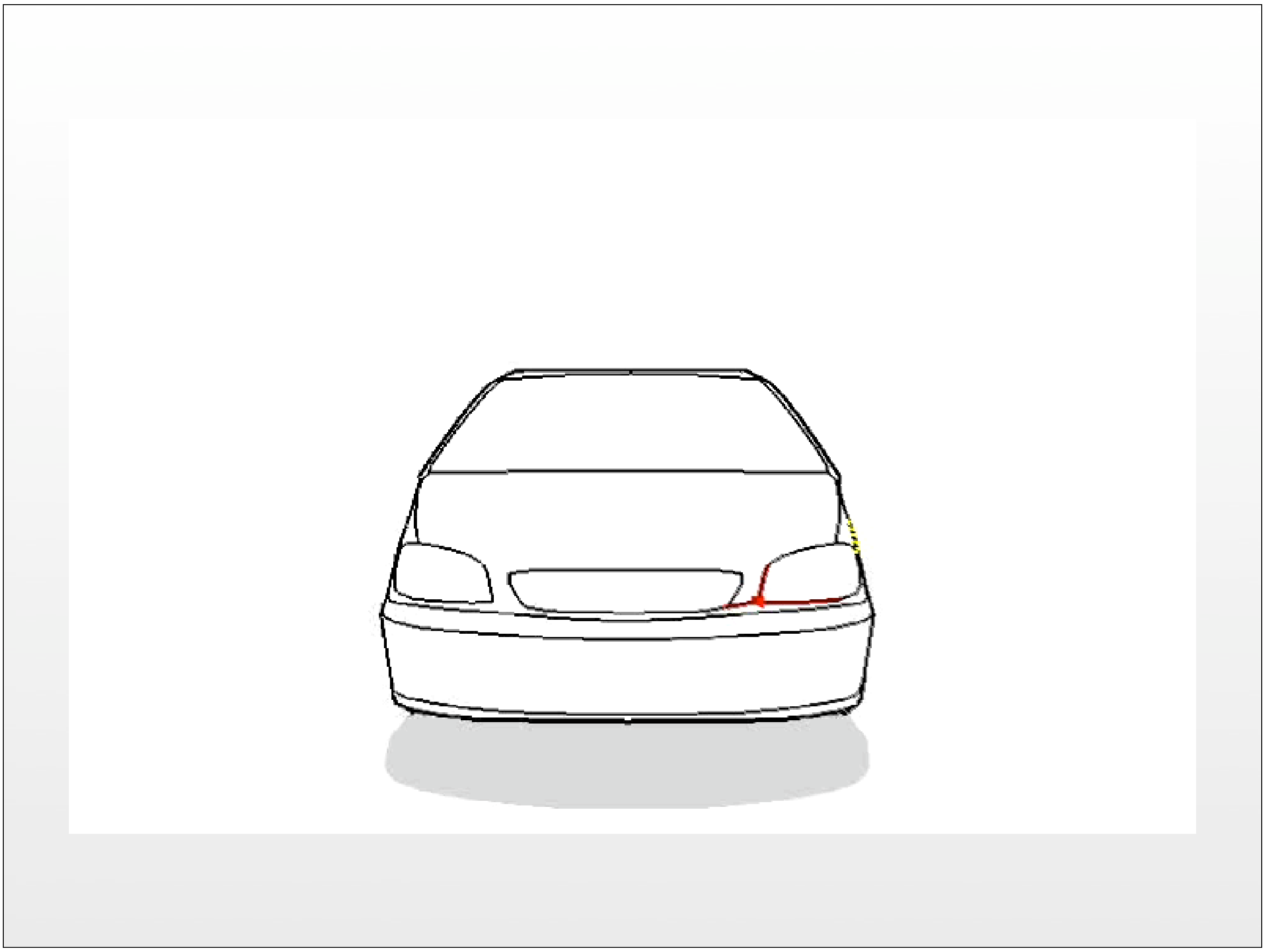


12

…

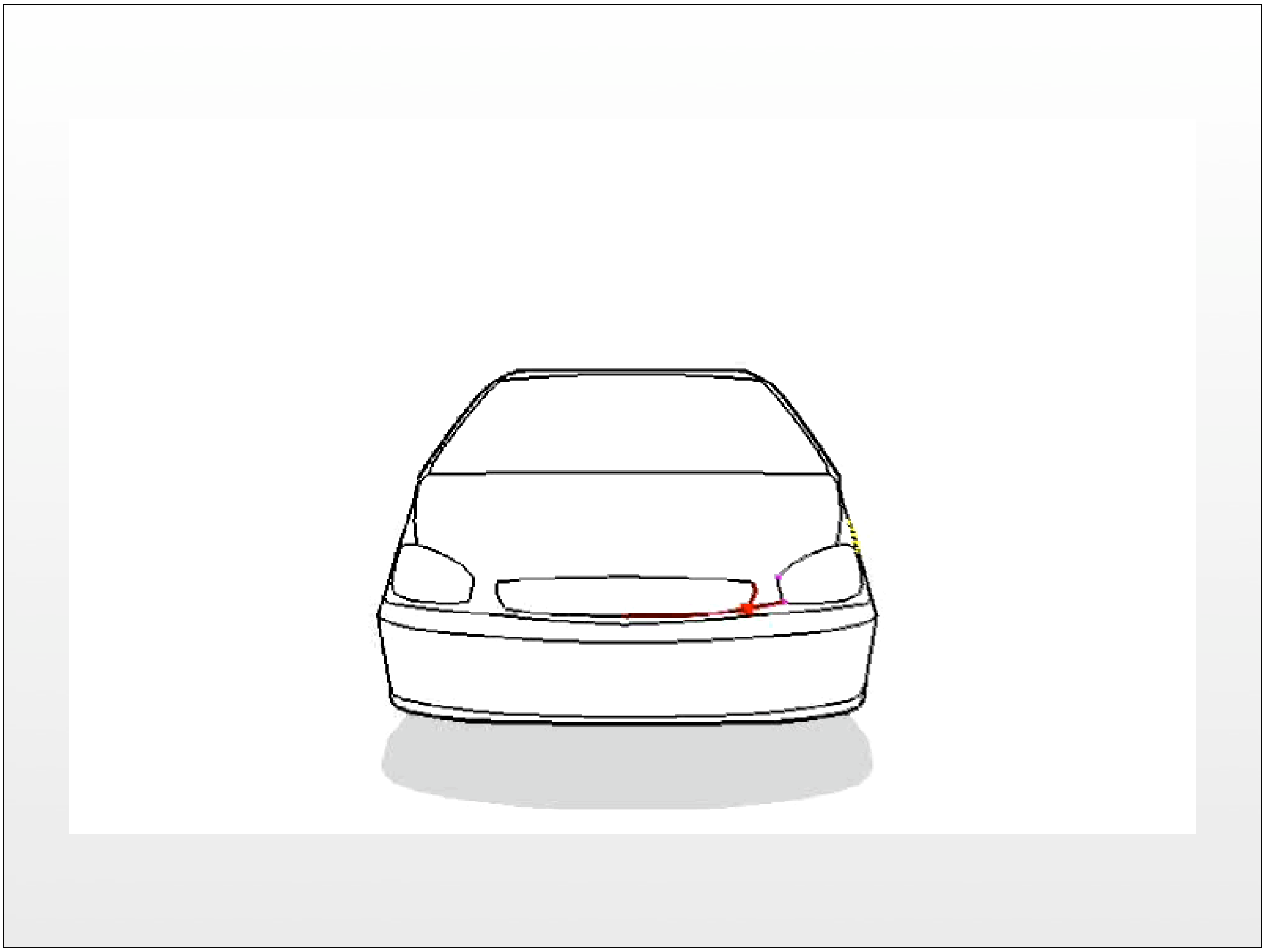


And he can always evaluate the shape from different viewpoints ... and continue working on the shape ...



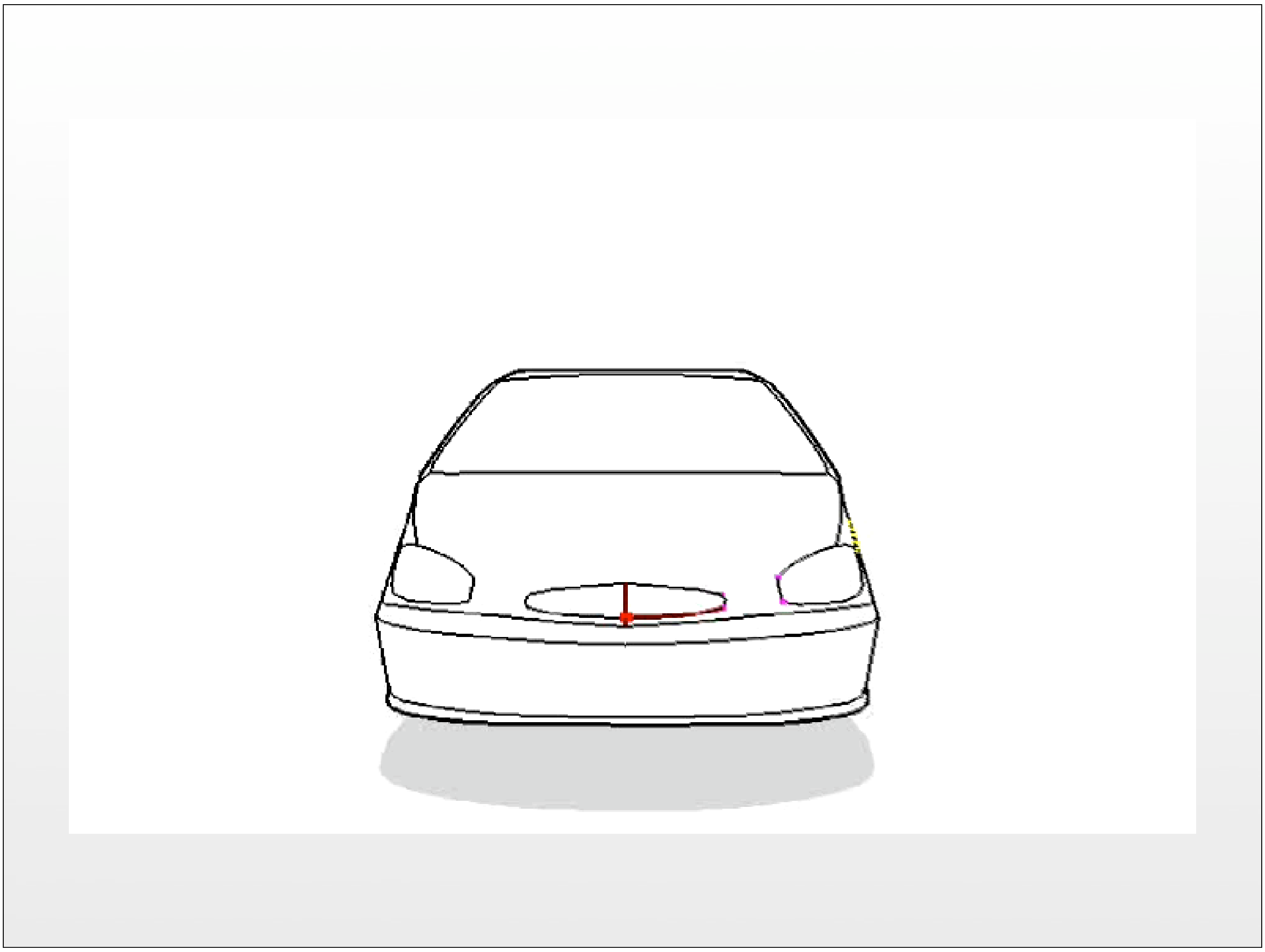
14

in the new view, as we demonstrate here by modifying the headlights ...



15

... and the grill, again the process is fast, intuitive and explorative.



16

And upon finishing we will have 3d representation ready.

How?

So the question is how to implement this system. There are two aspects to this:

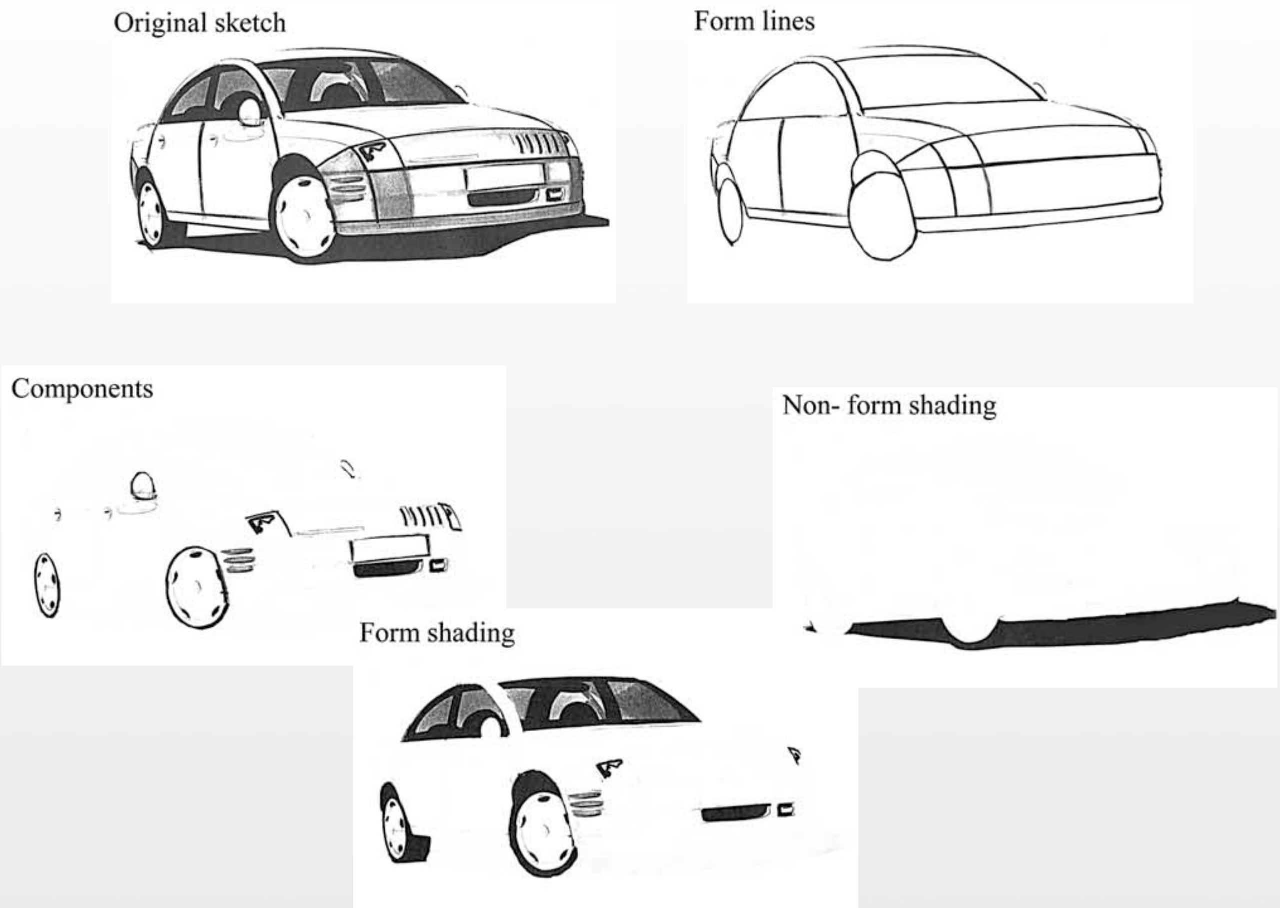
first we have to find a suitable representation for car shapes

second we have to find a way to modify it

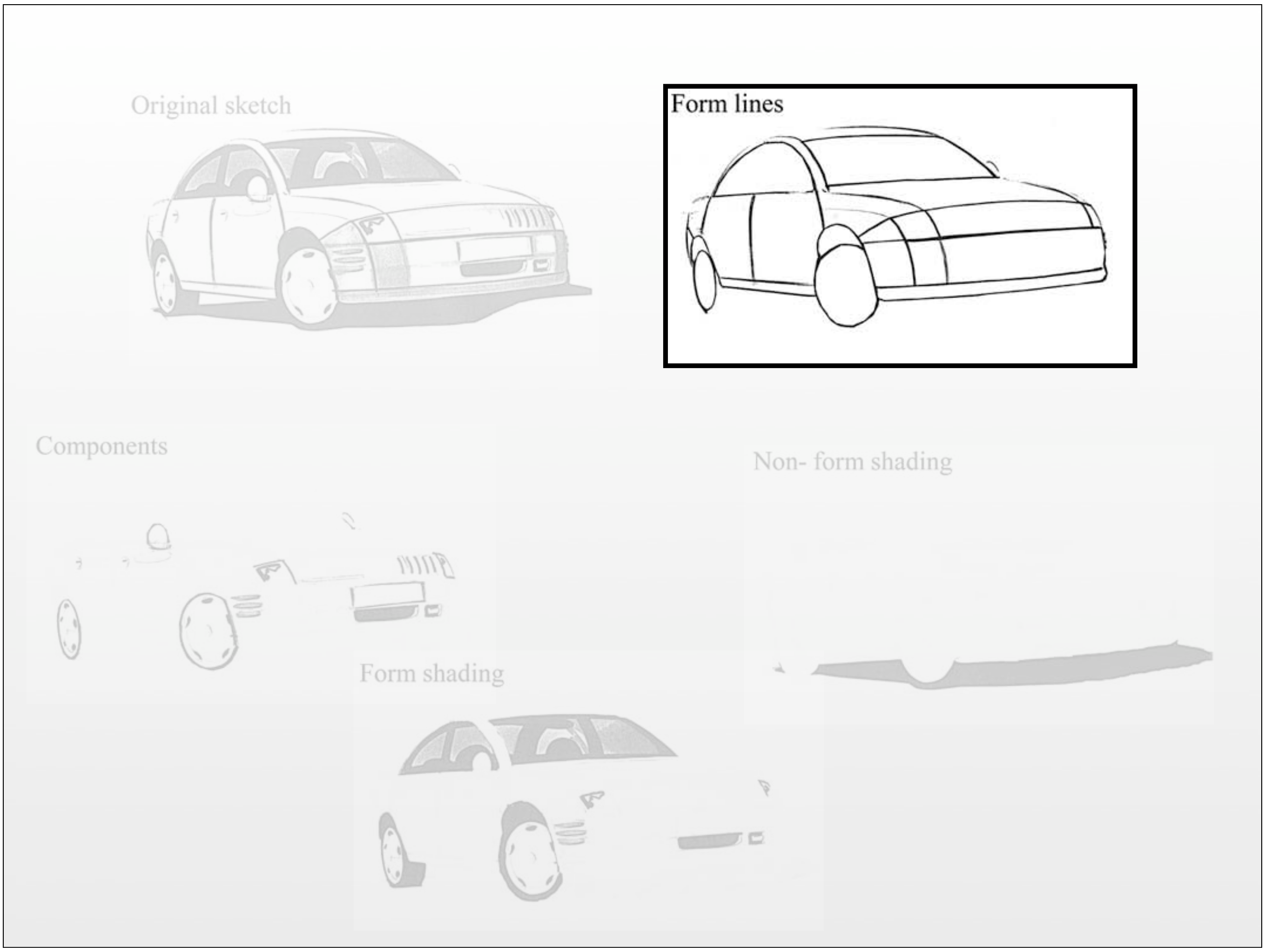
Representation

18

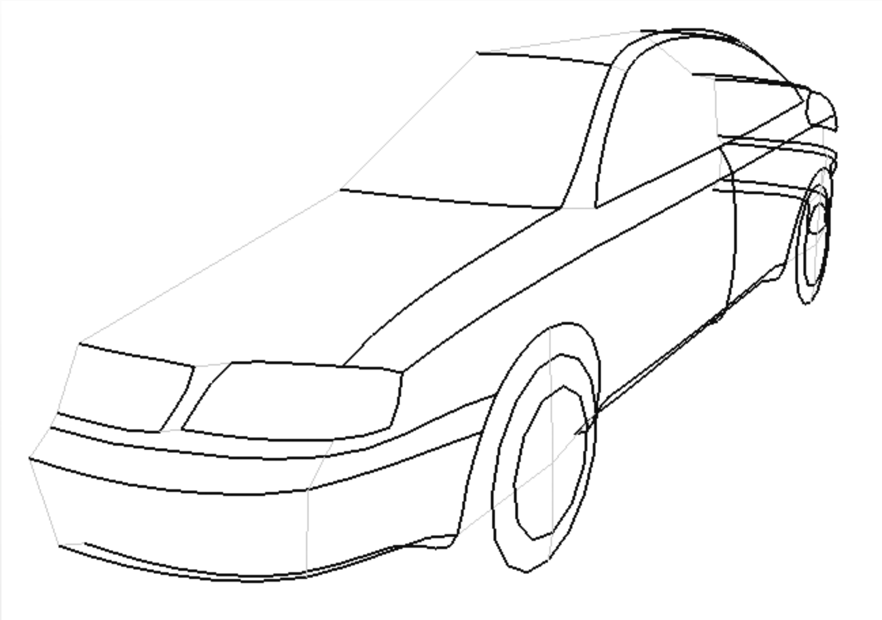
first the representation. We have to find one which is natural for designers to work with, and describe the important characteristics of the shape well.



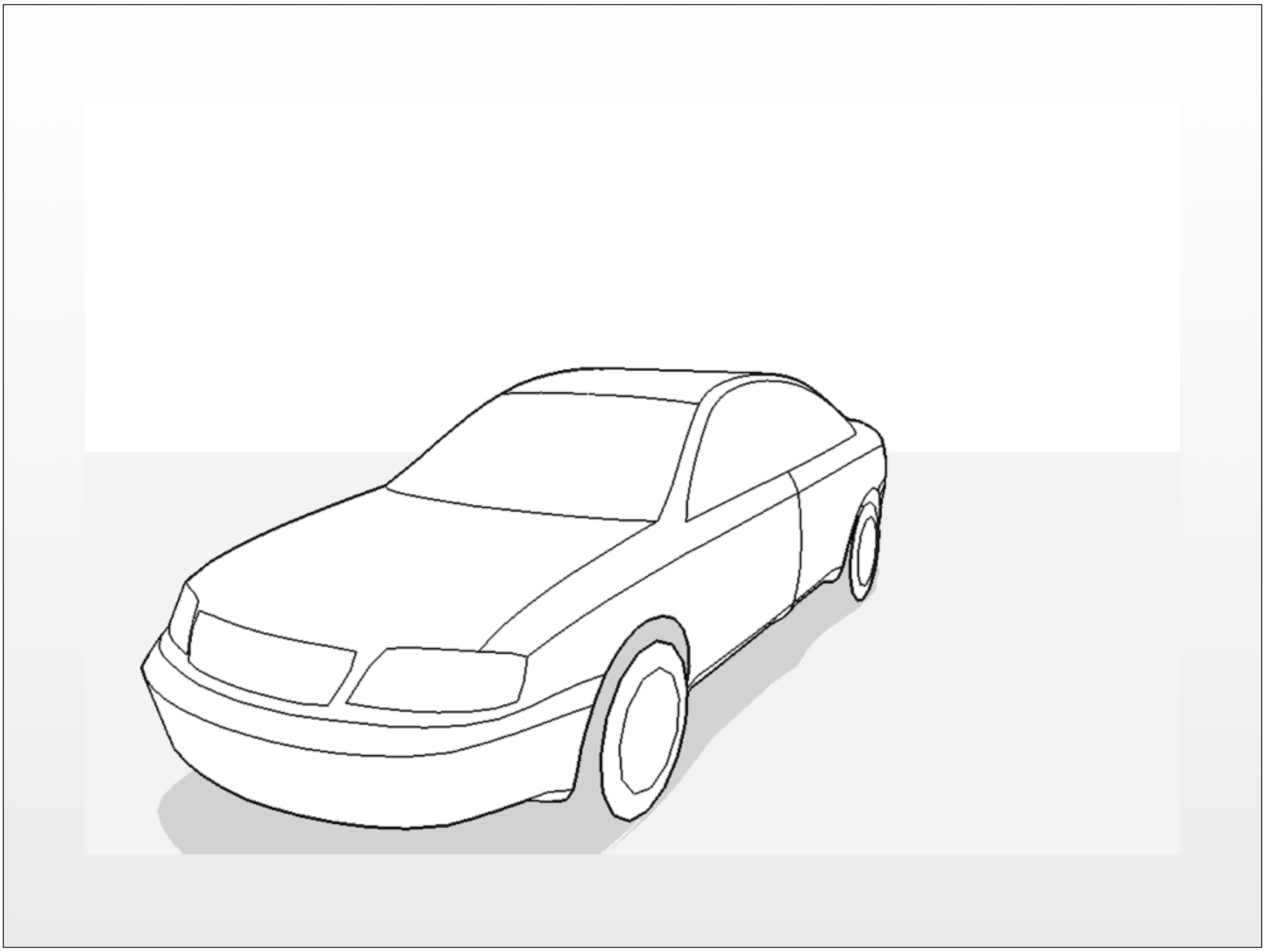
Fortunately others before us already studied the anatomy of a car sketch, and how designers draw. And defined the different levels of a sketch...



We will concentrate on the most important, on the form lines, which convey the most information about the shape, and get drawn first by all designers. We created a polyline network to represent a fairly generic set of form lines...



... you can see it here, notice we use only half a car because of the symmetry. We also associated with the polyline network a triangle mesh, so we can do...



22

hidden line removal for a nice rendering. As we are sticking to our topology during the process (at least in this work) the only free variables in our representation are the vertex coordinates.

$\mathbf{x}$

So the vector x of the vertex coordinates will be the one which we will modify, for which value we will be looking for during...

Modifying the shape

... modifying the shape. There are two aspects to modifying the shape, first we have to satisfy the constraints set by the user and also we have to make sure that we always working with a car shape...

$$\arg\min_{\mathbf{x}} E_s + E_c$$

$$\|\mathbf{P}\mathbf{x} - \mathbf{x}_c\|$$

25

So we will be minimizing two term simultaneously one for the shape and one for the constraints...
The term for the constraint is relatively easy. For the pulling/dragging of points we need to define the 2d positions of the constrained points (xc) and make sure that the L2 distance between those positions and the corresponding positions coming from the model are minimized. In the above formula P represents the current projective transformation and also select only the points from x that are constrained.
Implementing the sketch over feature is uses the same optimization, but we first have to establish a correspondence between the drawn line and our representation. We do this in 2d with the help of snakes. I unfortunately don’t have time to go into the details now.

$$\arg\min_{\mathbf{x}} E_s + E_c$$

26

So now for the other term, which is a bit more involved. This term should tell us if we are still working on a car shape or we just satisfied the constraints but what we have is a rabbit and not a car.

So how to define this term. First we will define a term which will tell us if we are similar to a given shape. In that way we can choose an initial shape for modeling, and define our goal to stay somehow similar to the shape and satisfy the constraints. For that we have to define a similarity measure first. The question is where should we measure similarity. The easy idea of measuring the similarity between the x vectors are not really working, we have to define a mapping ...

$$\mathbf{f} = \mathbf{\Phi}(\mathbf{x})$$

$$E_s(\mathbf{x}) = \|\mathbf{\Phi}(\mathbf{x}) - \mathbf{f}\|$$

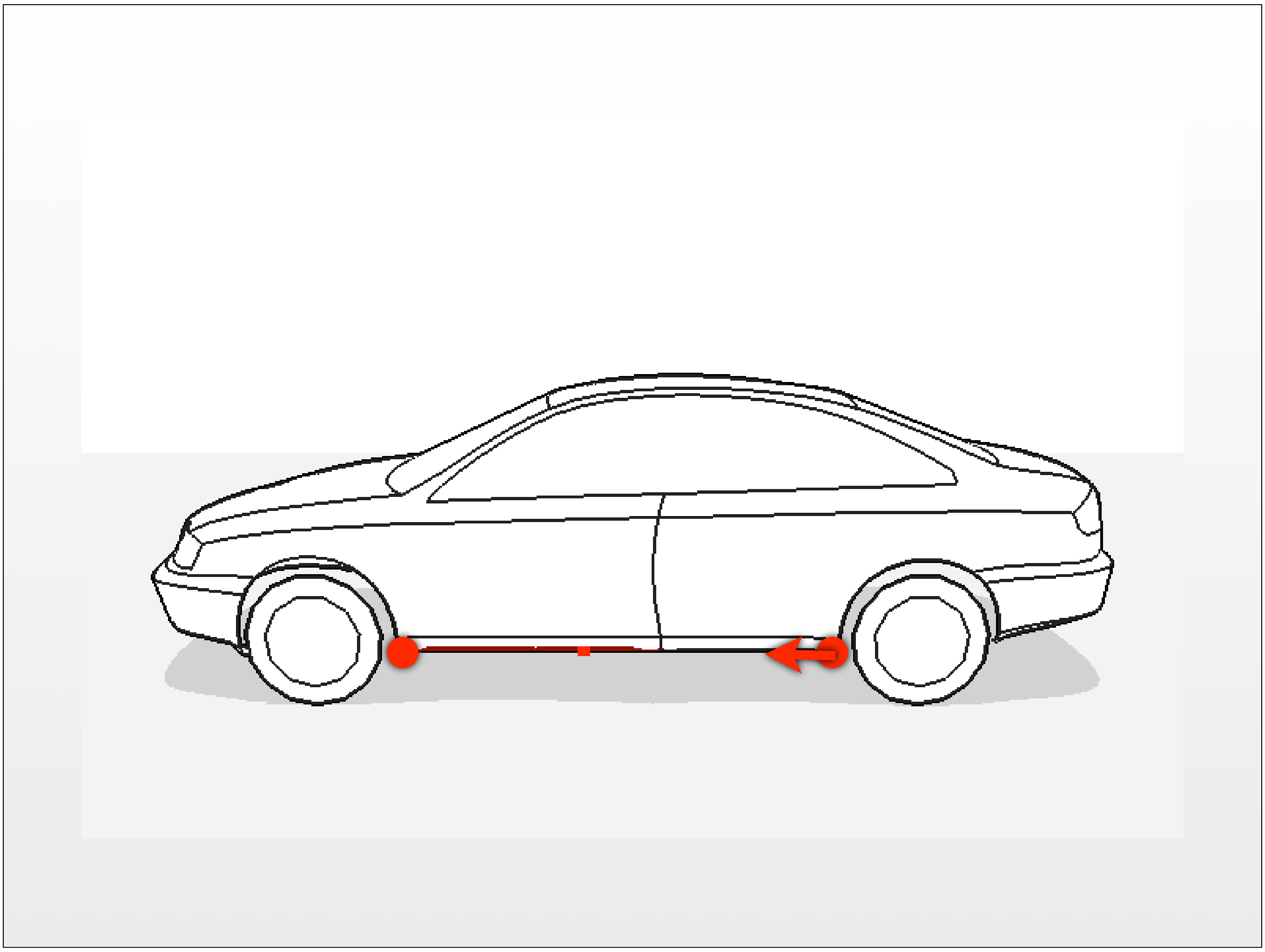
... phi which maps our x to a suitable feature vector and then if we have an initial shape with a known f, than we can define our shape term like this. So the only question that remains, is how to choose phi?

This is an interesting research topic on its own, unfortunately again, i don't have time to go into the details, but we need something which is a local property, like curvature or normals of the surface. In the paper we used an approximation for deformation gradients. For now it is enough that most of these features can be calculated from x with a linear operator G

$$\mathbf{f} = \mathbf{G}\mathbf{x}$$

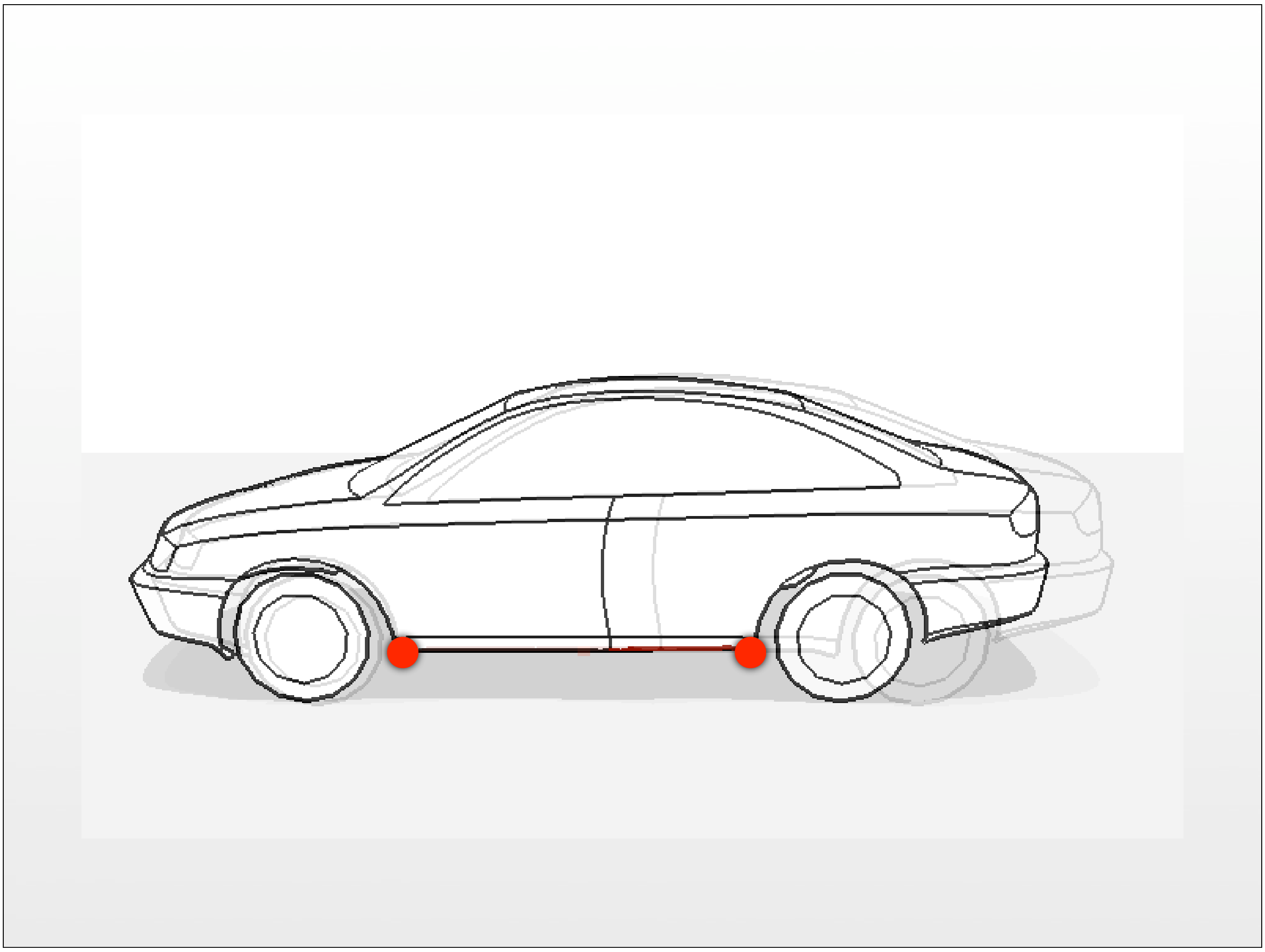
$$E_s(\mathbf{x}) = \|\mathbf{G}\mathbf{x} - \mathbf{f}\|$$

... so the first equation becomes f = Gx and the second one Gx – f, which makes our optimization process relatively easy. Now I’d like to show the effect of this...



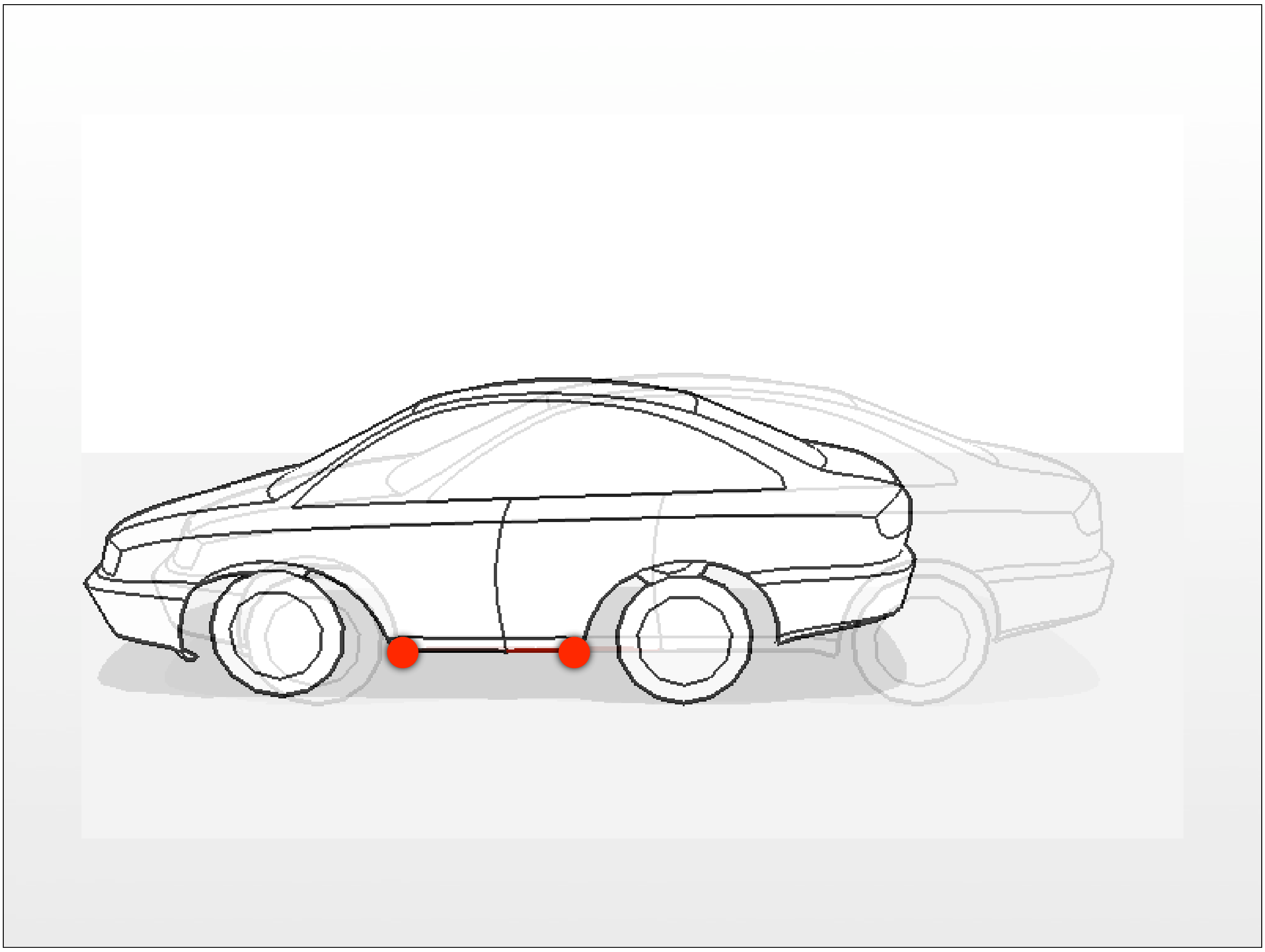
29

So we have our initial model ... we choose some points to constrain .. and I’ll just move one of them a bit...



30

the result satisfies the new constraints, looks like a car and is very similar ... to the starting point
This is all nice, but if we would like two do bigger changes, the system falls apart...



31

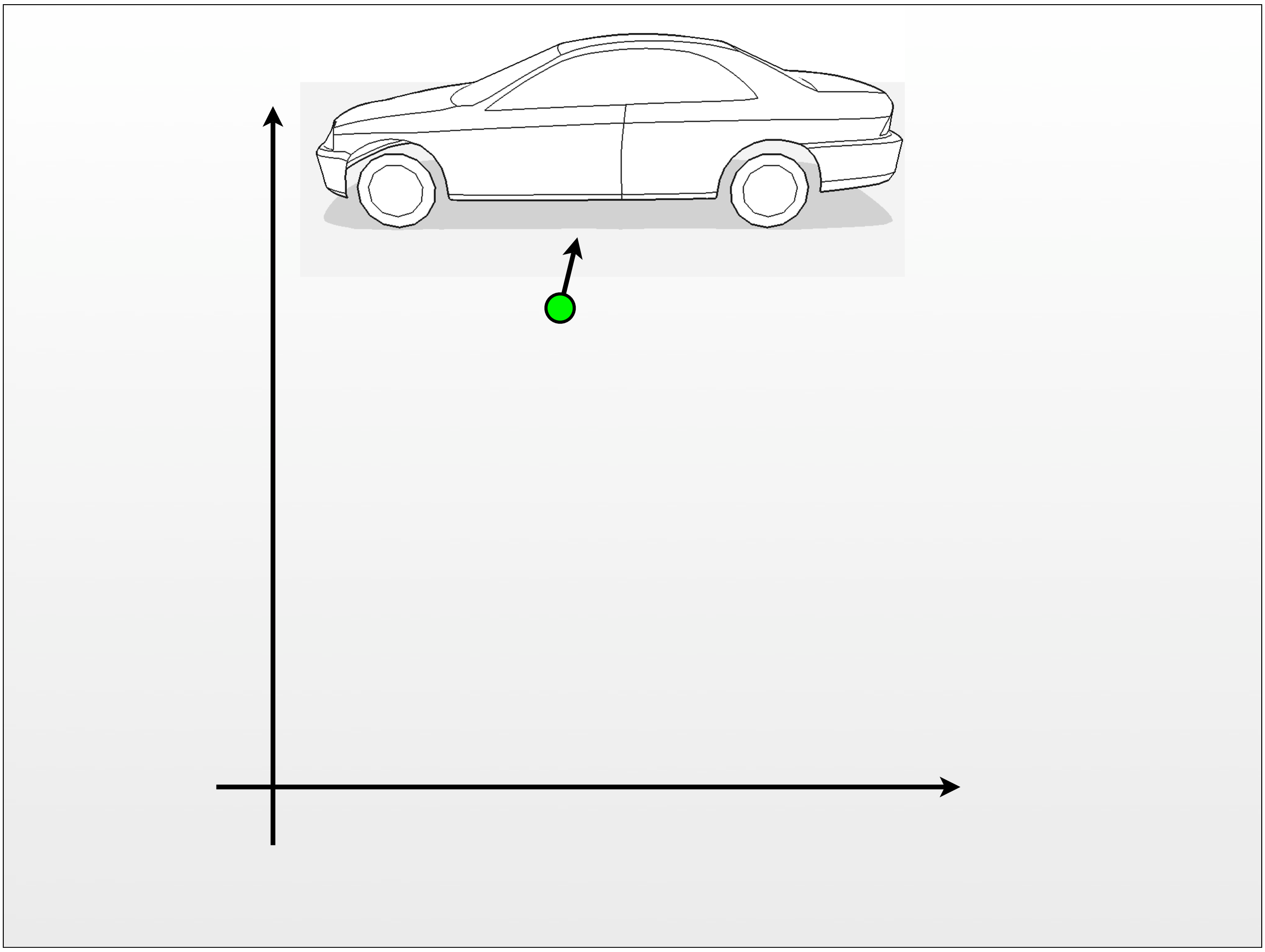
as you can see in this example ... This is because our similarity measure is only tries to maintain local similarity and has no knowledge whatsoever about what a car is. We can still use this and constrain more points to get a better shape, but it is not very intuitive anymore.

$$E_s(\mathbf{x}) = \|\mathbf{G}\mathbf{x} - \mathbf{f}\|$$

We concentrated in the previous slides on how to define G, to get good similarity measure, now we have to change f from a fixed value to a dynamic one. We need a model for f...

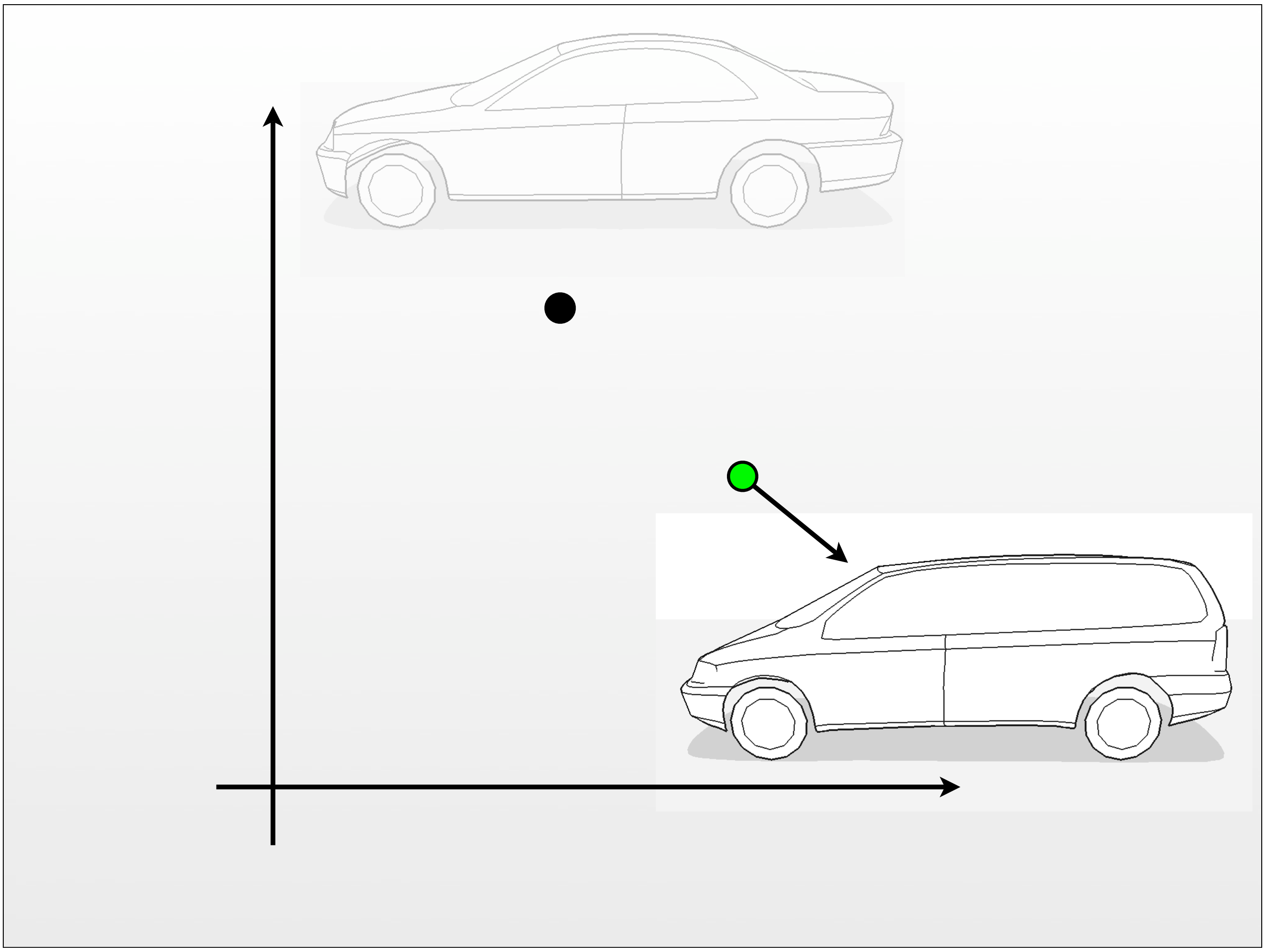
$$E_s(\mathbf{x}, \mathbf{w}) = \|\mathbf{G}\mathbf{x} - \mathbf{f}(\mathbf{w})\|$$

Where f is dependent on some parameter vector, and in our optimization we will optimize this w alongside x.
To define this model for f we are using real life example shapes.



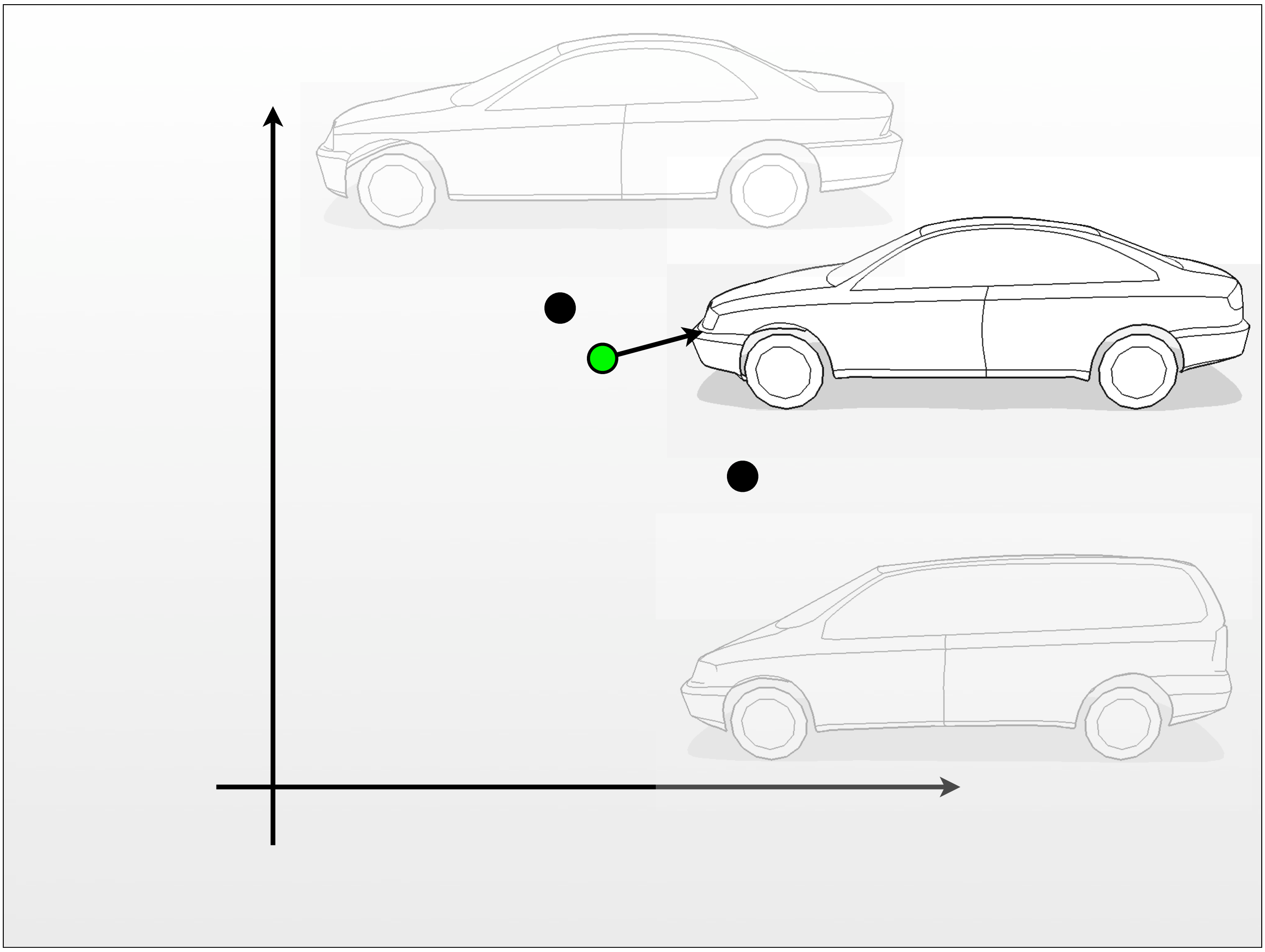
34

So we converted a real car to our representation and find the corresponding feature value f for it...

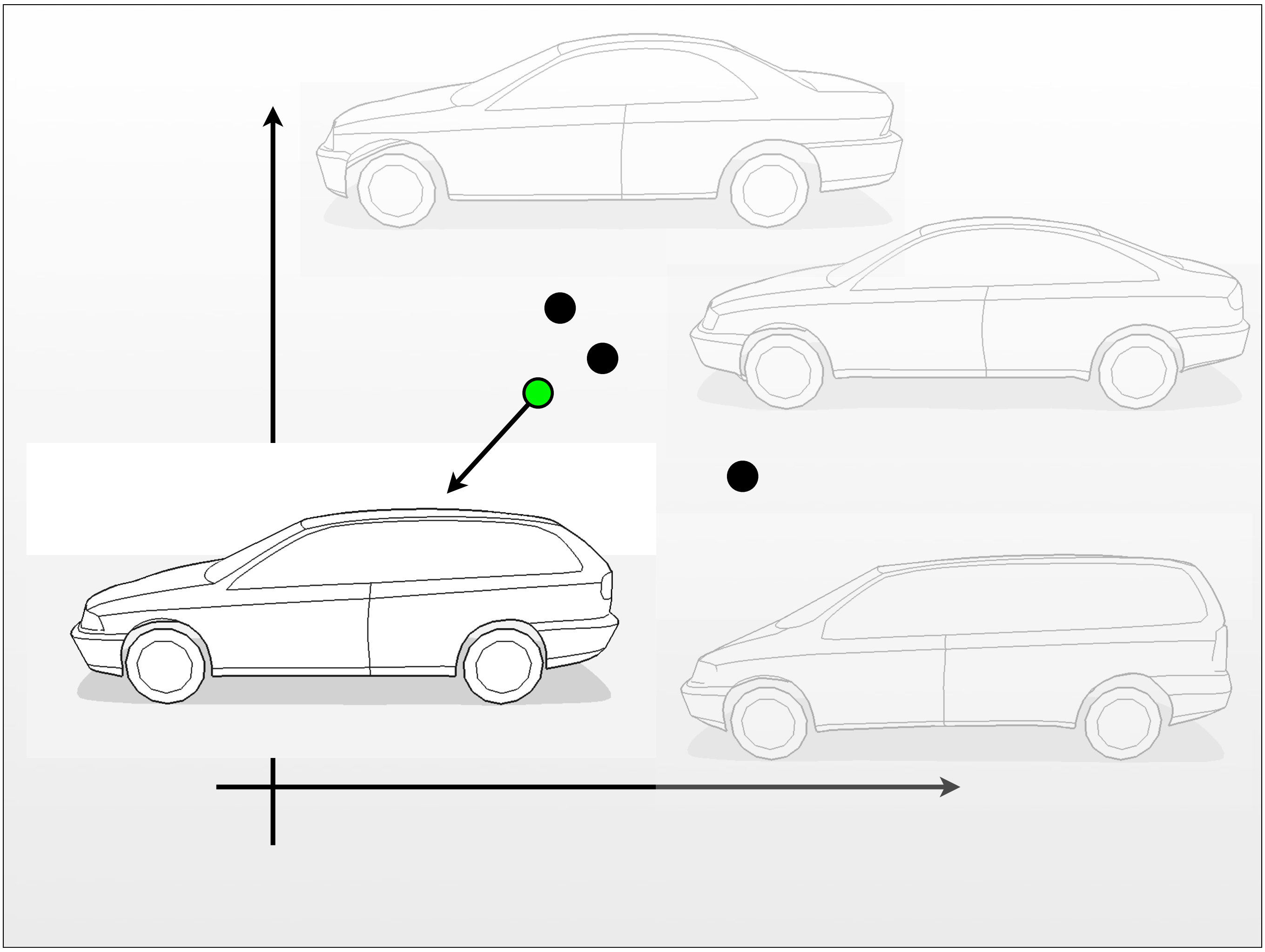


35

then another one

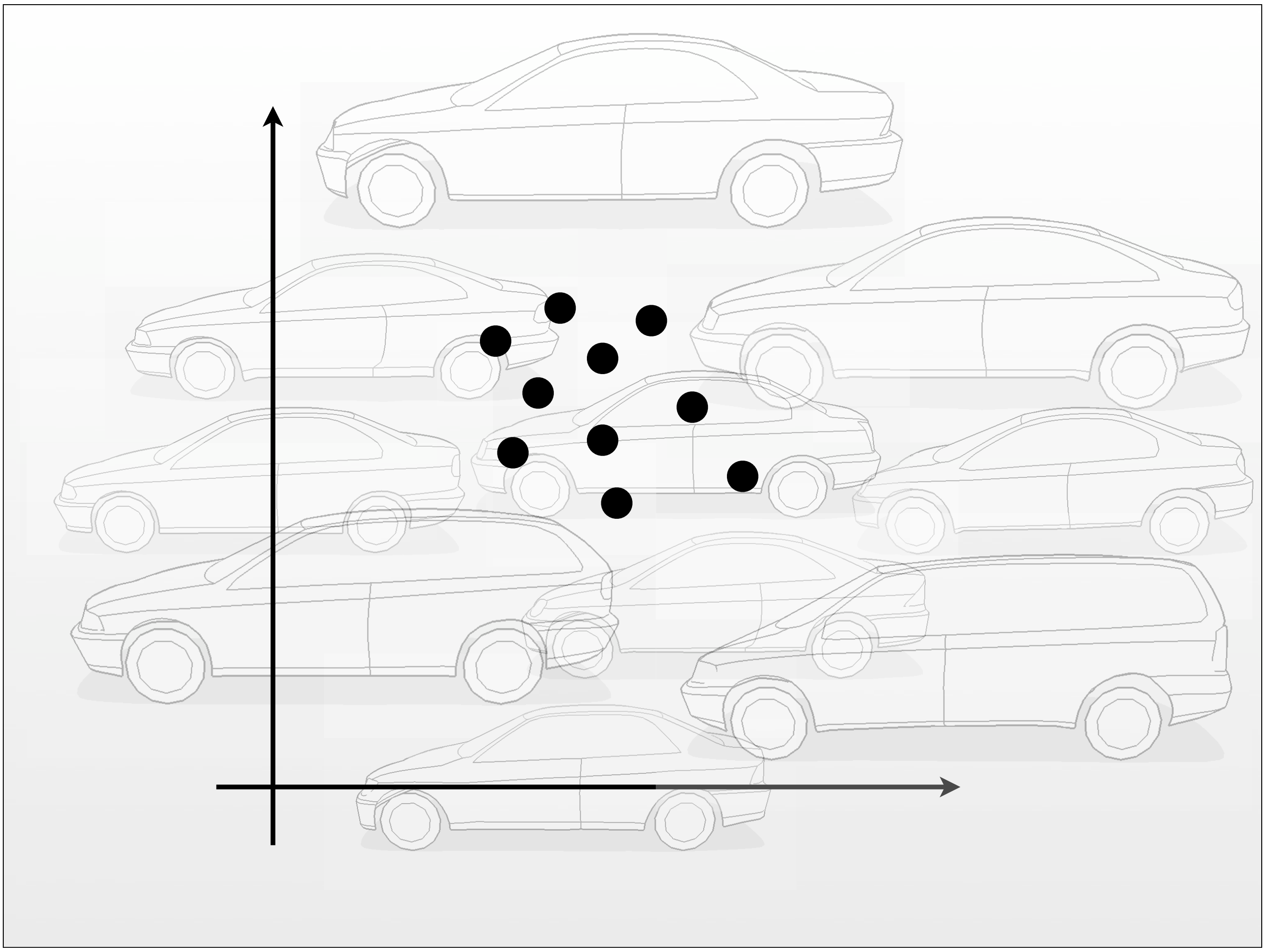


and another one

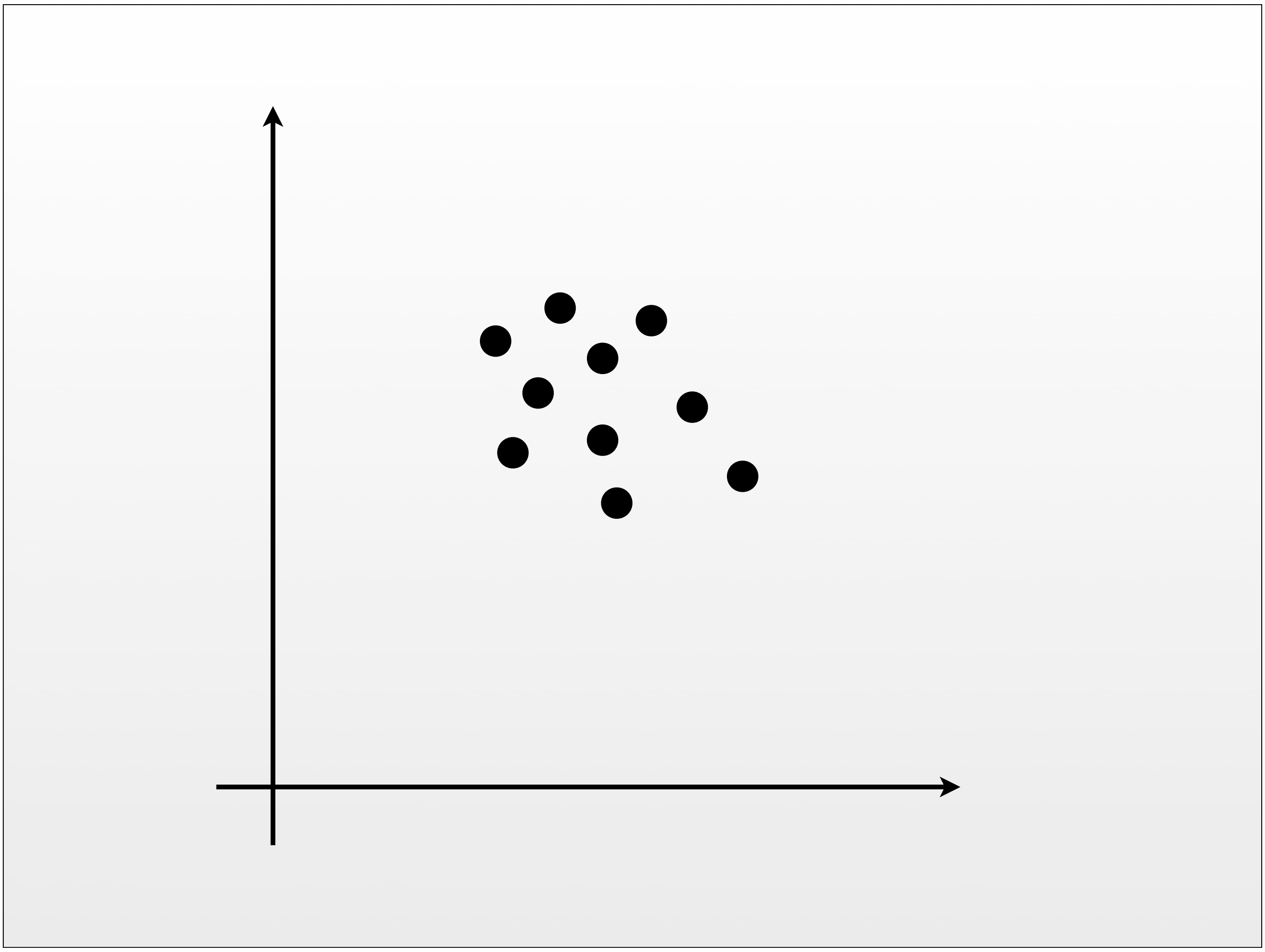


37

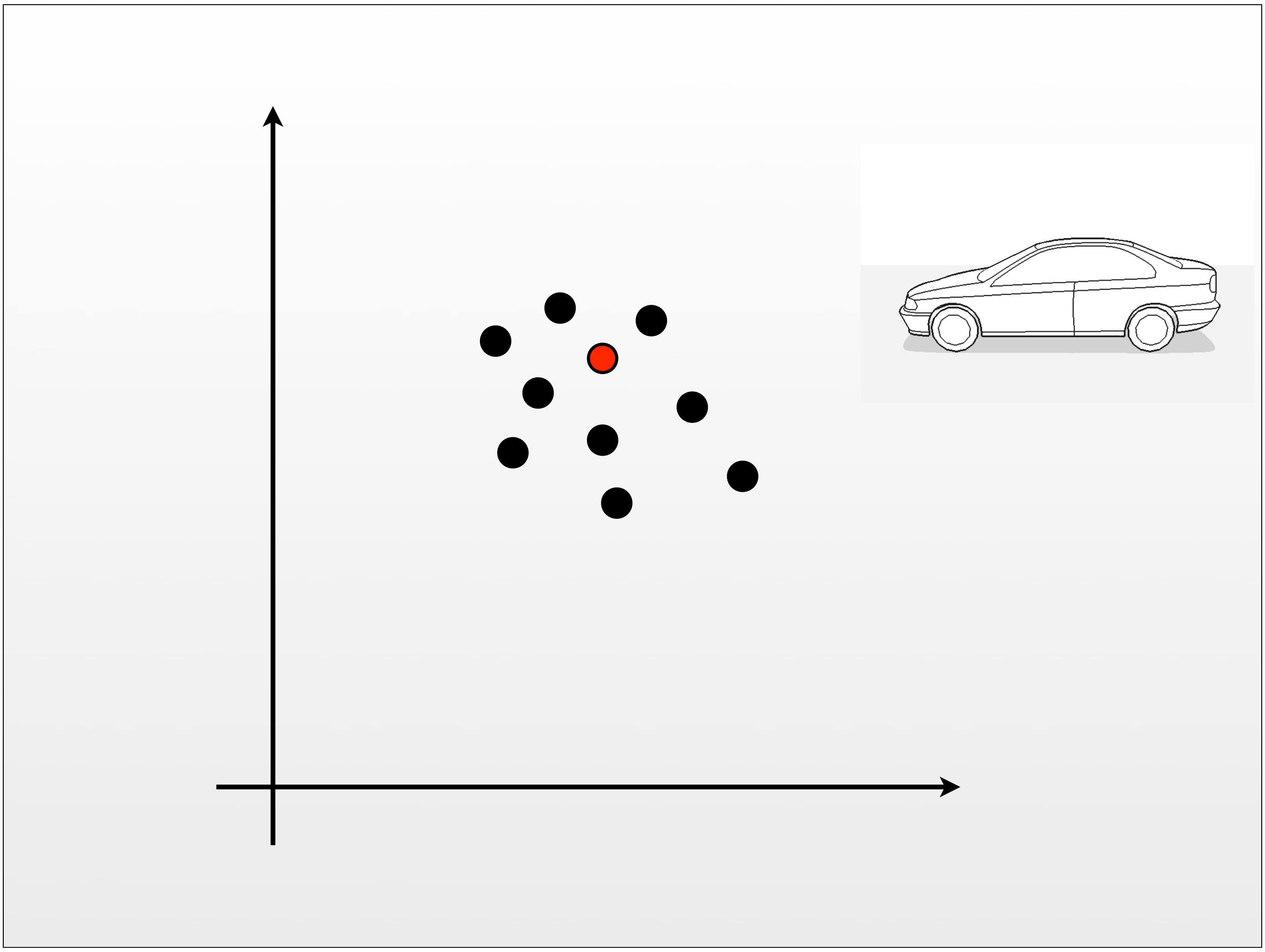
and one more



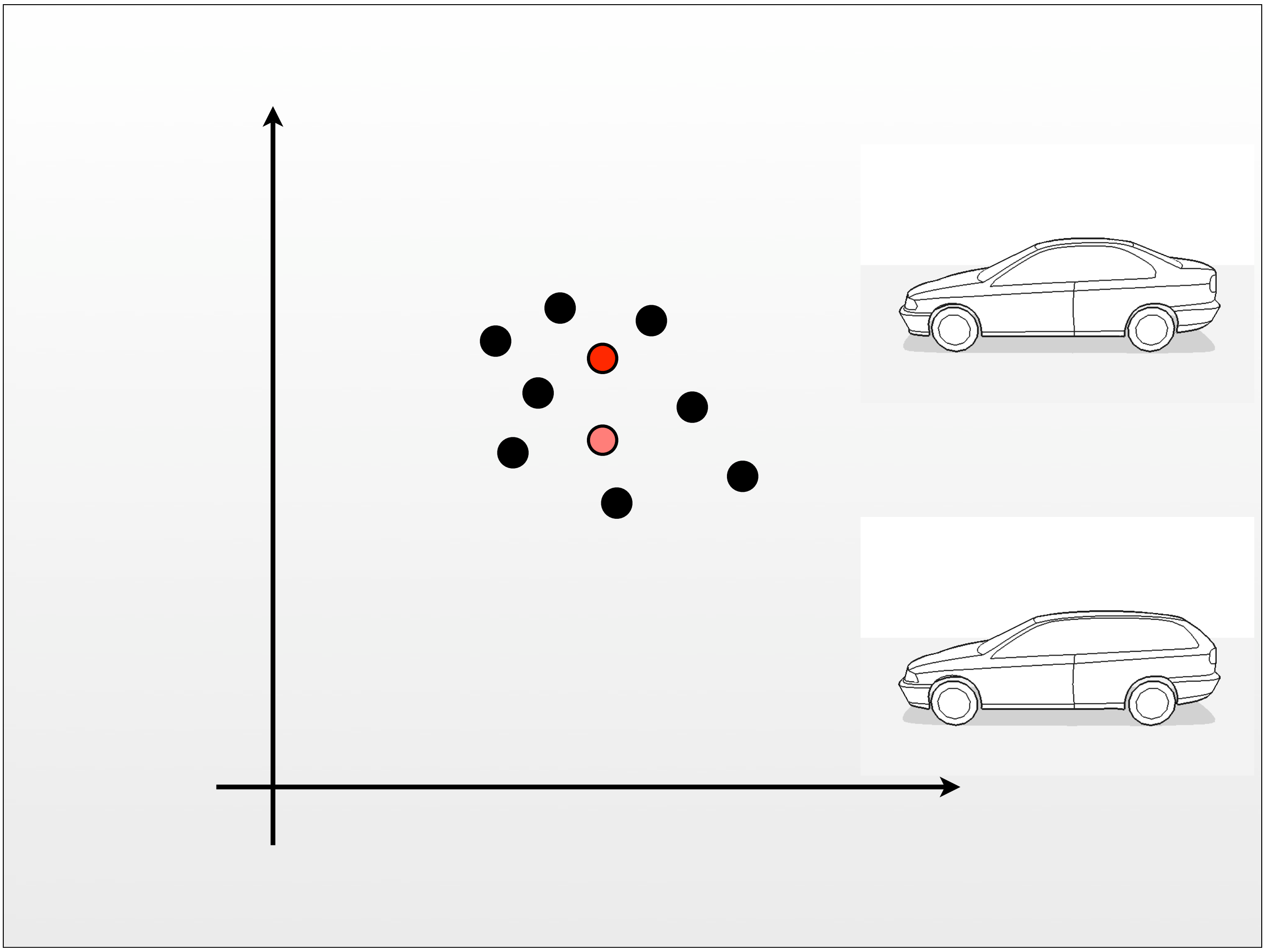
until we had a few examples...



now we can just calculate a linear combination of all the examples and generate new shapes, we have to be careful though, because the linear model extrapolates very poorly as I’ll demonstrate in the following slides.

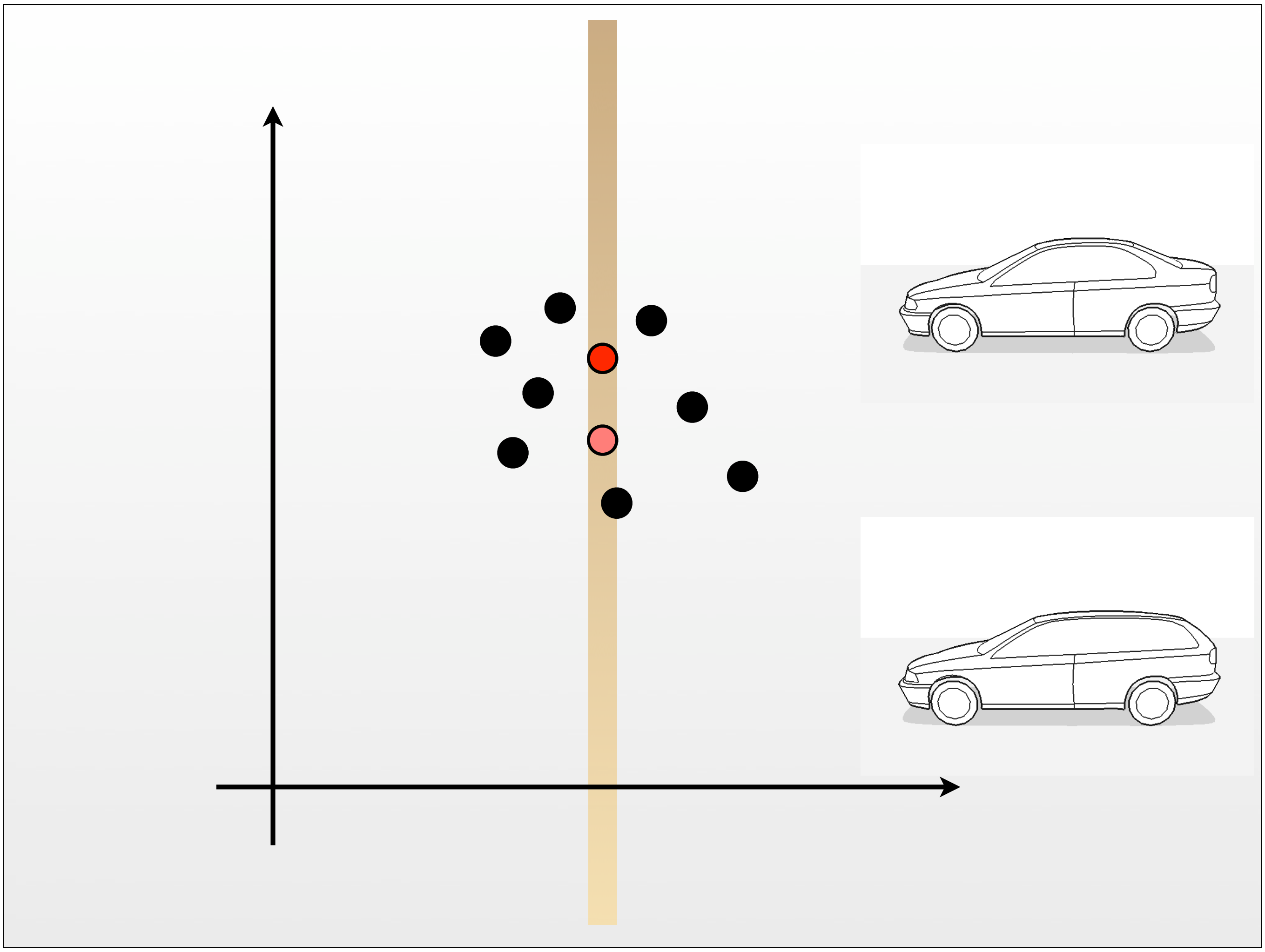


so take one example

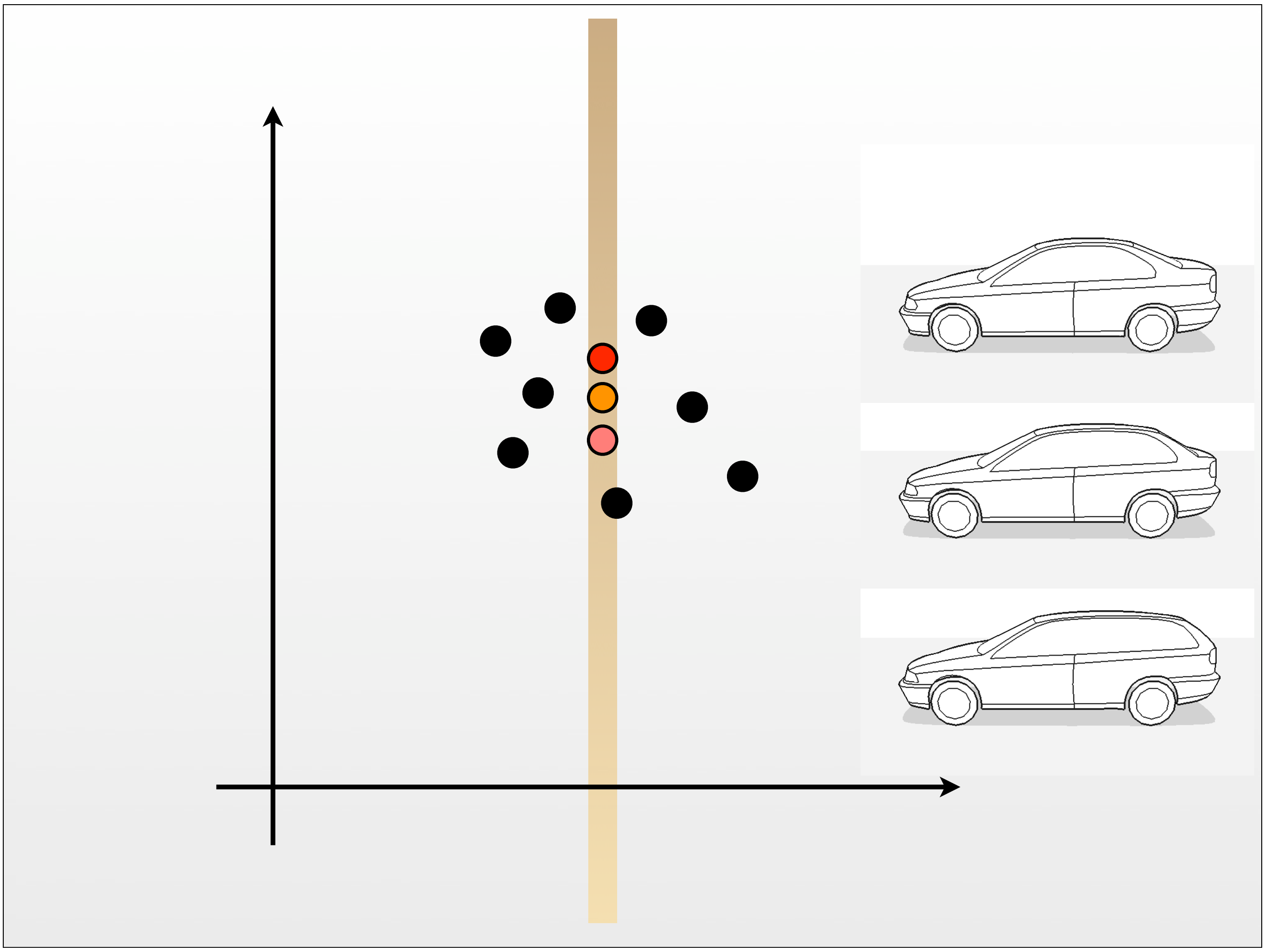


41

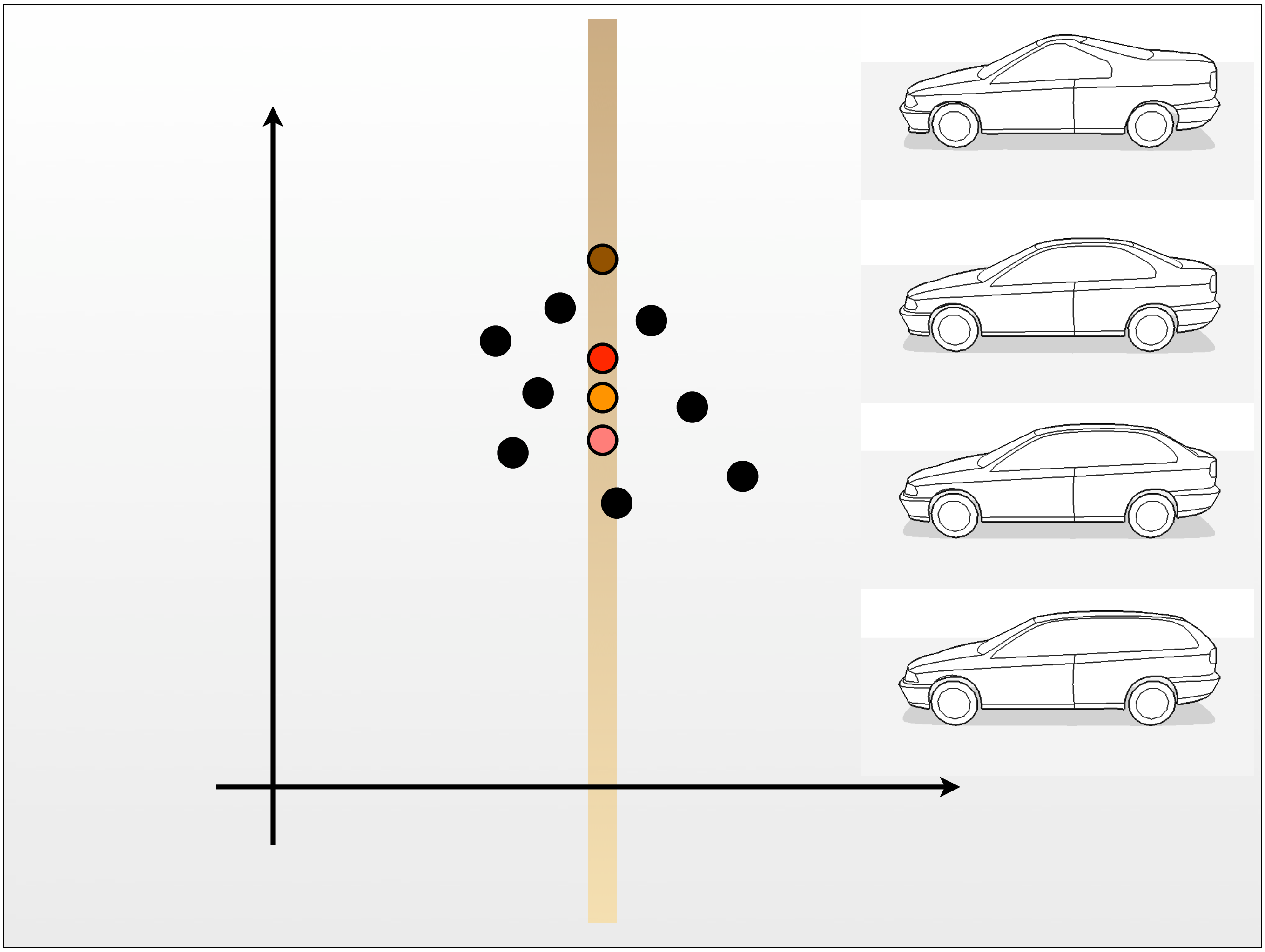
and another



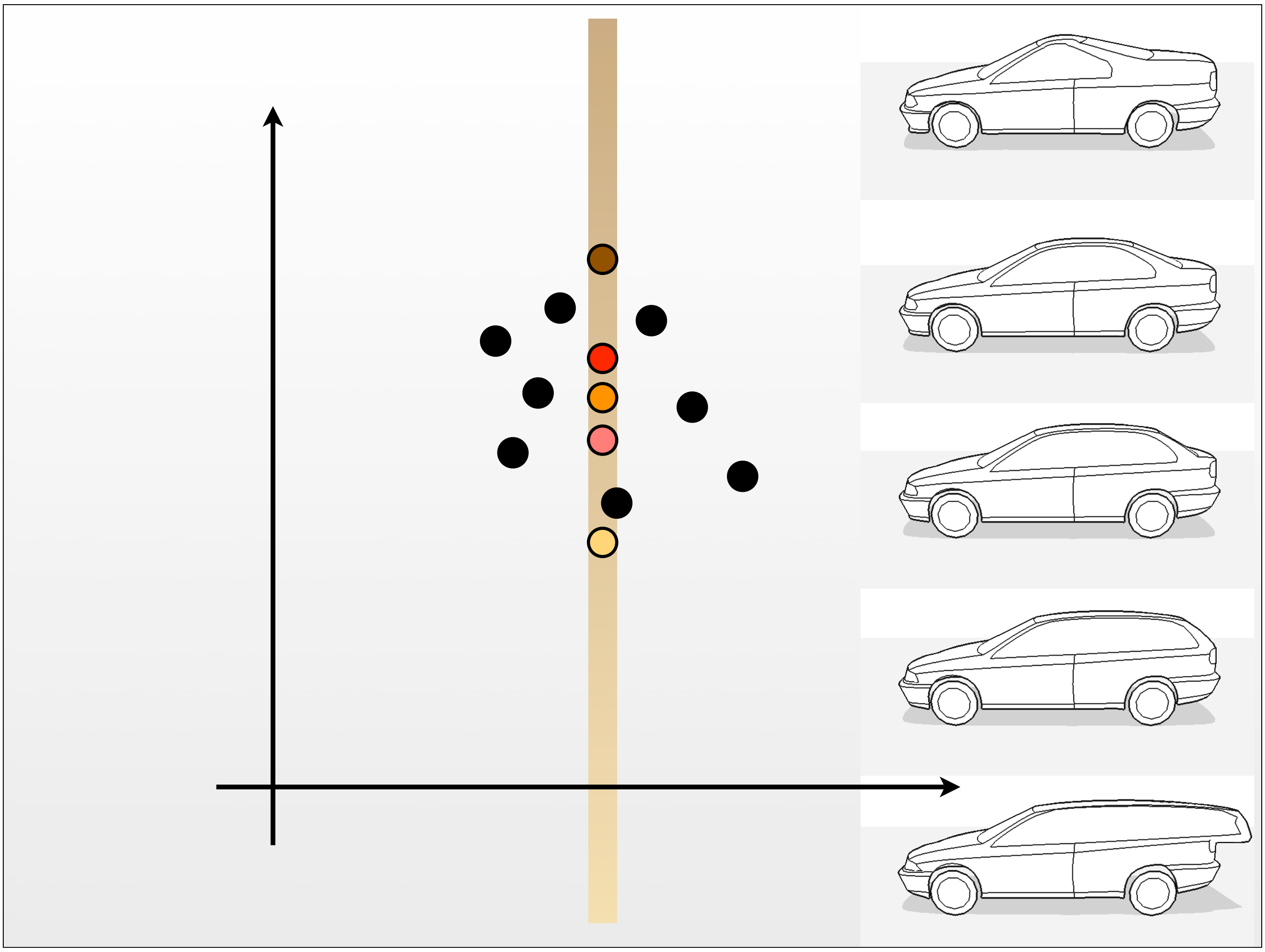
we can calculate their linear combination and it works



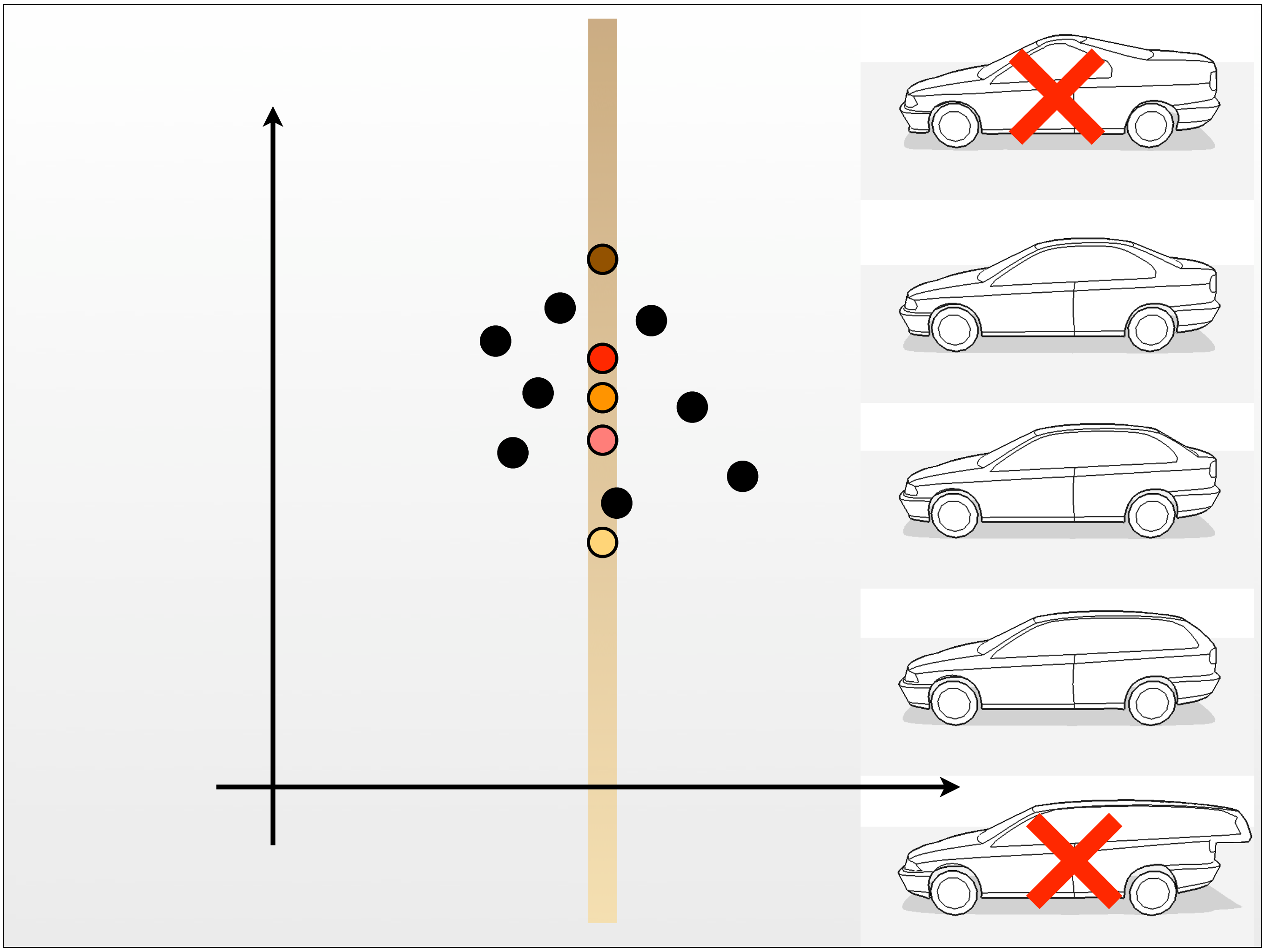
when we are interpolating between the two meshes



but starts to fall apart when we are extrapolating

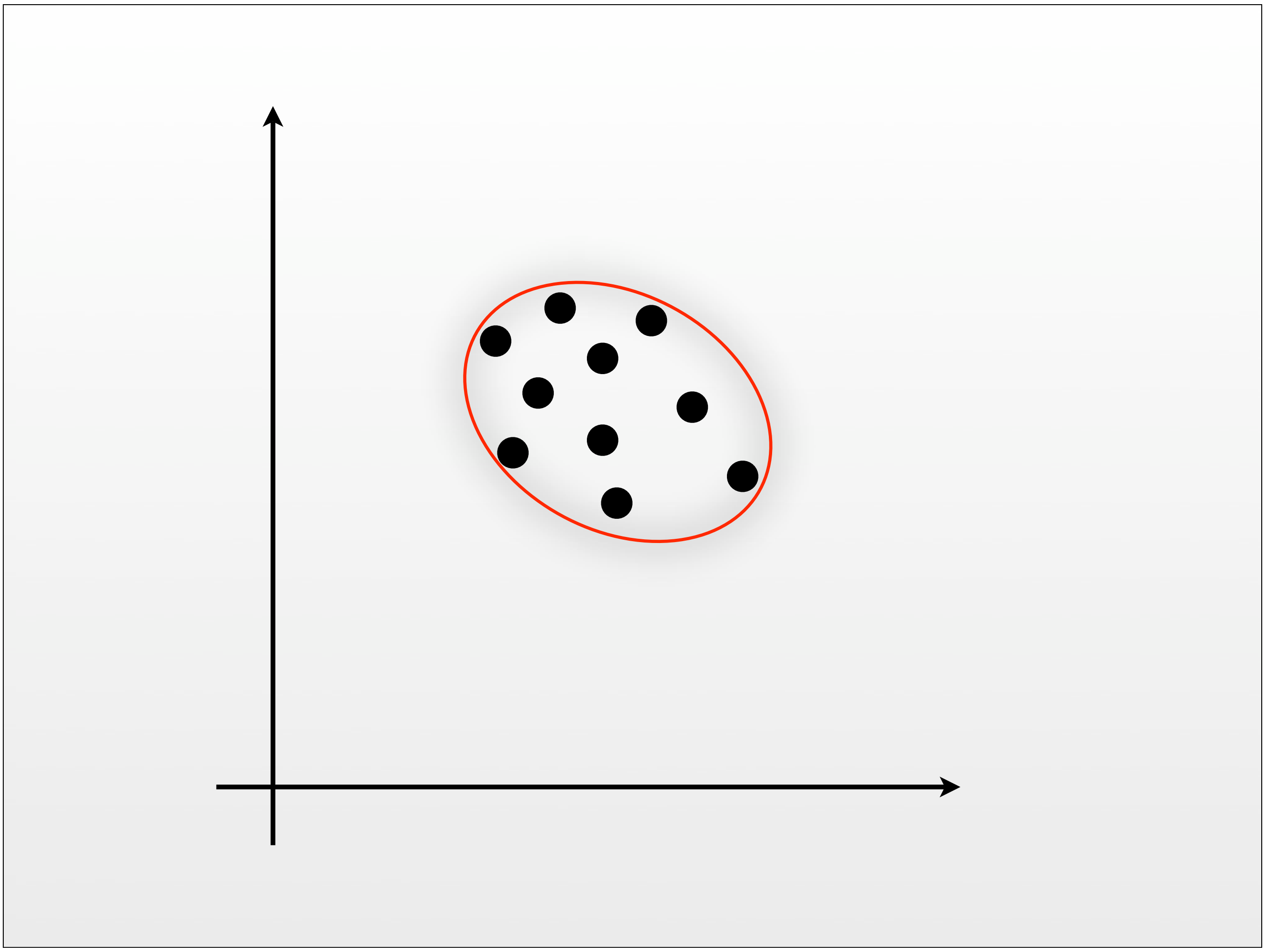


as you can see here better...

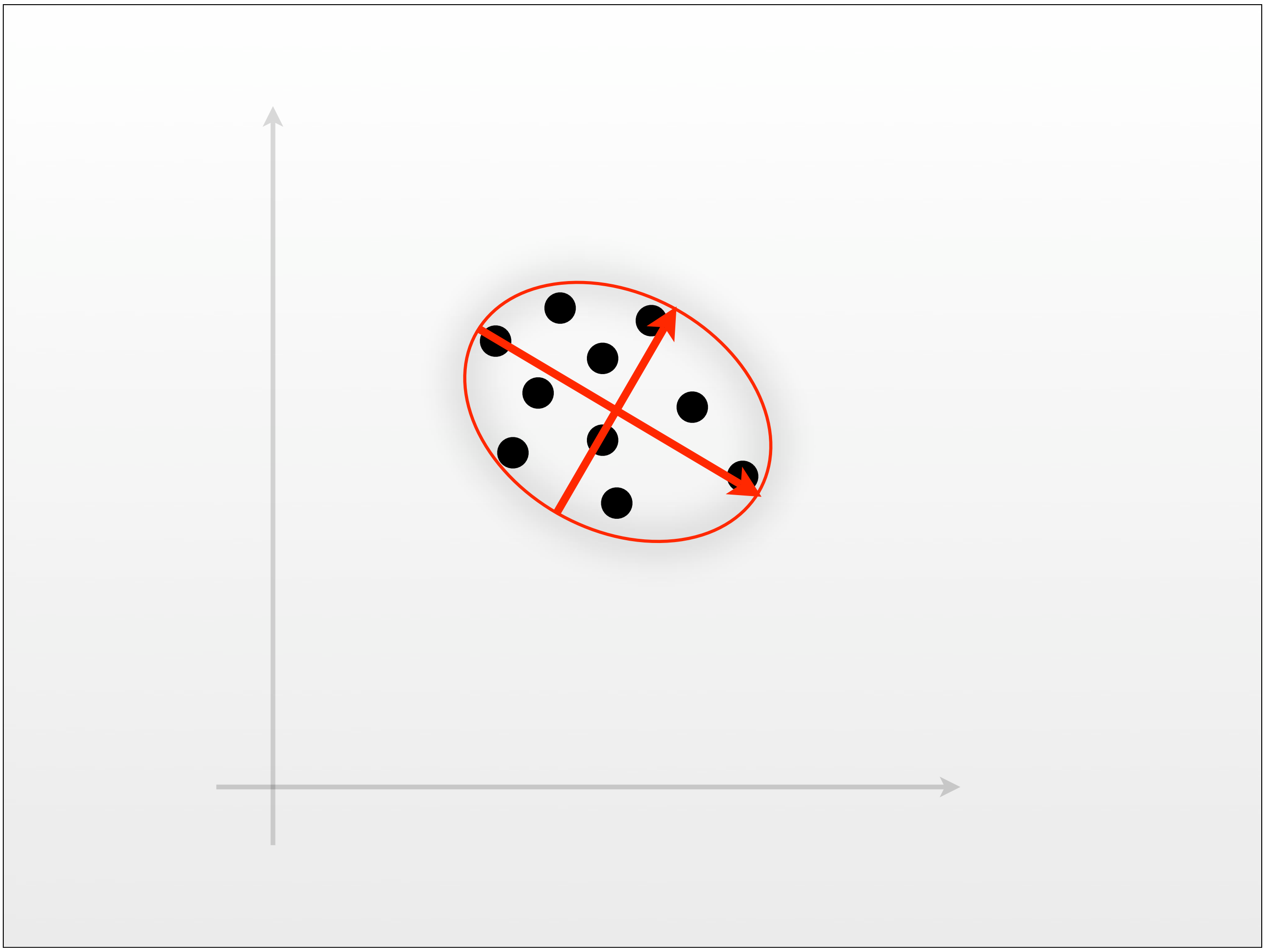


46

So we can only use the shapes within the convex hull of the examples



... we can also reparameterize this space by calculating the principal axes of the example distribution ...



48

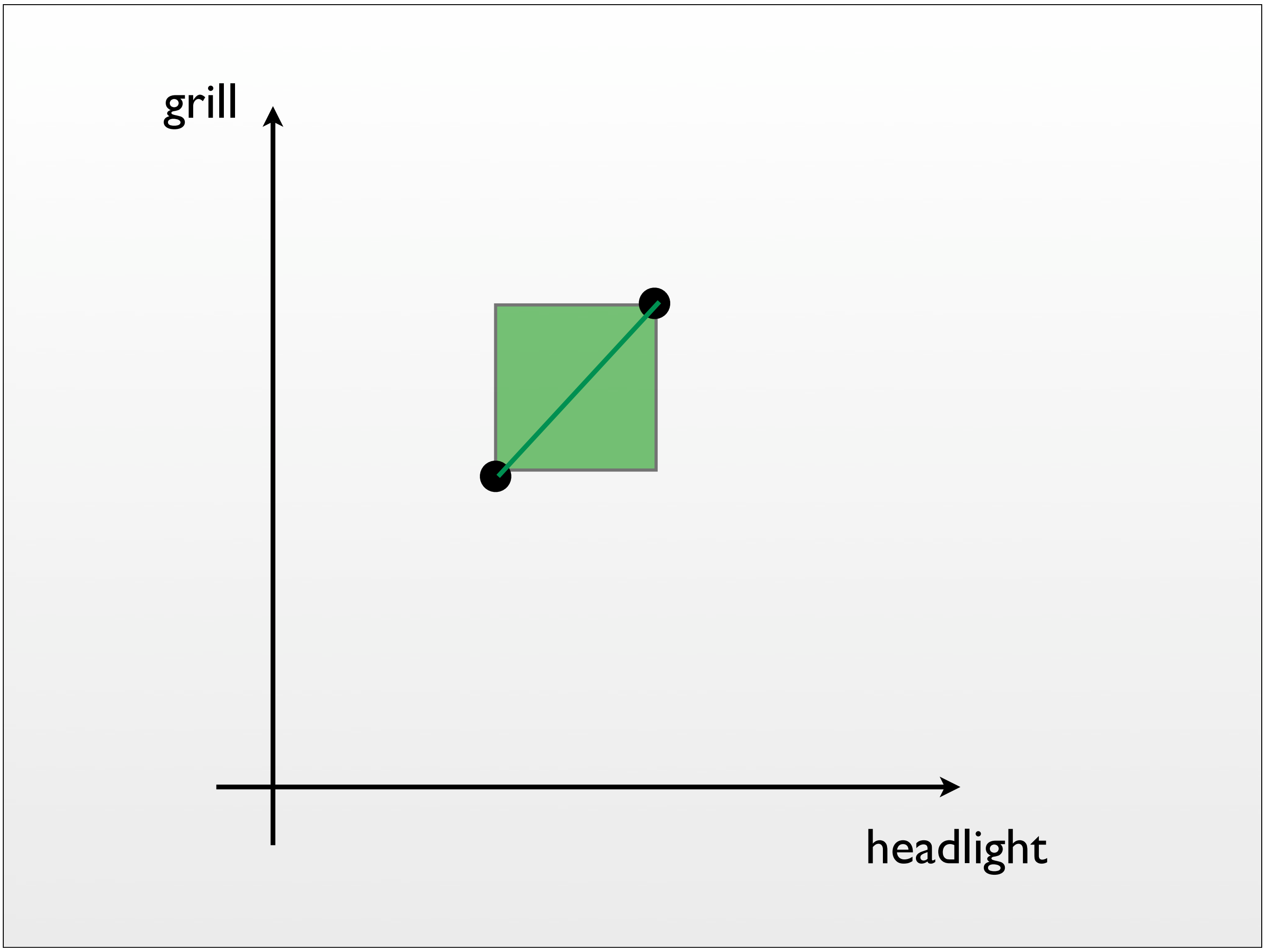
and use those directions to define our linear model, which makes the numerical calculation more stable.

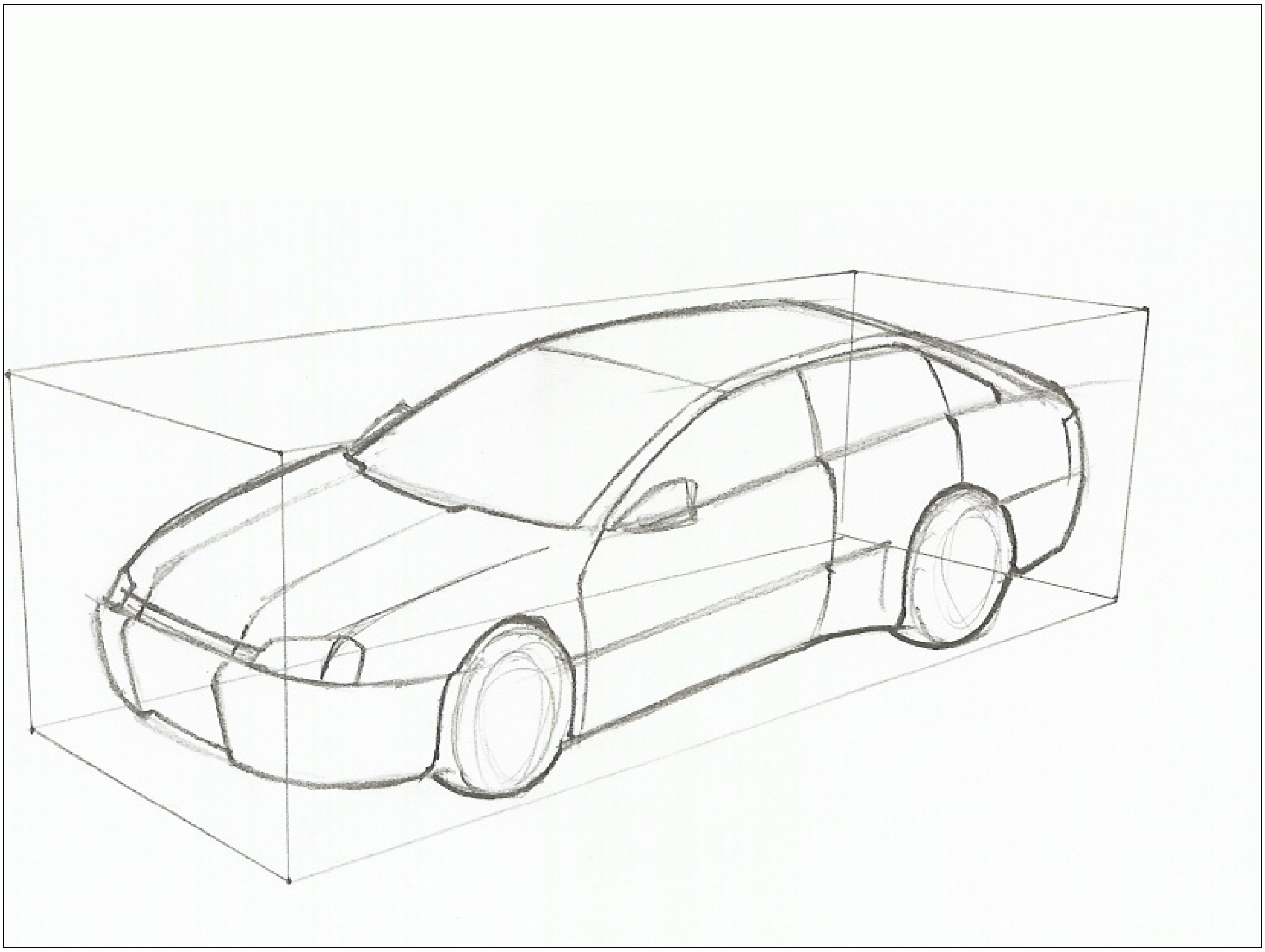
So that concludes my brief technical overview of our system,

$$\arg\min_{\mathbf{x},\mathbf{w}} \|\mathbf{G}\mathbf{x} - \mathbf{F}\mathbf{w}\| + \lambda\|\mathbf{P}\mathbf{x} - \mathbf{x}_c\| + \eta\|\mathbf{w}\|$$

Here I wrote again the complete optimization formula. As you can see we have Fw for the linear model of the features, and Gx for mapping x to the feature space, then we have the term for satisfying the constraints and the last term is for regularization, that is to get sure that we don’t extrapolate too much.

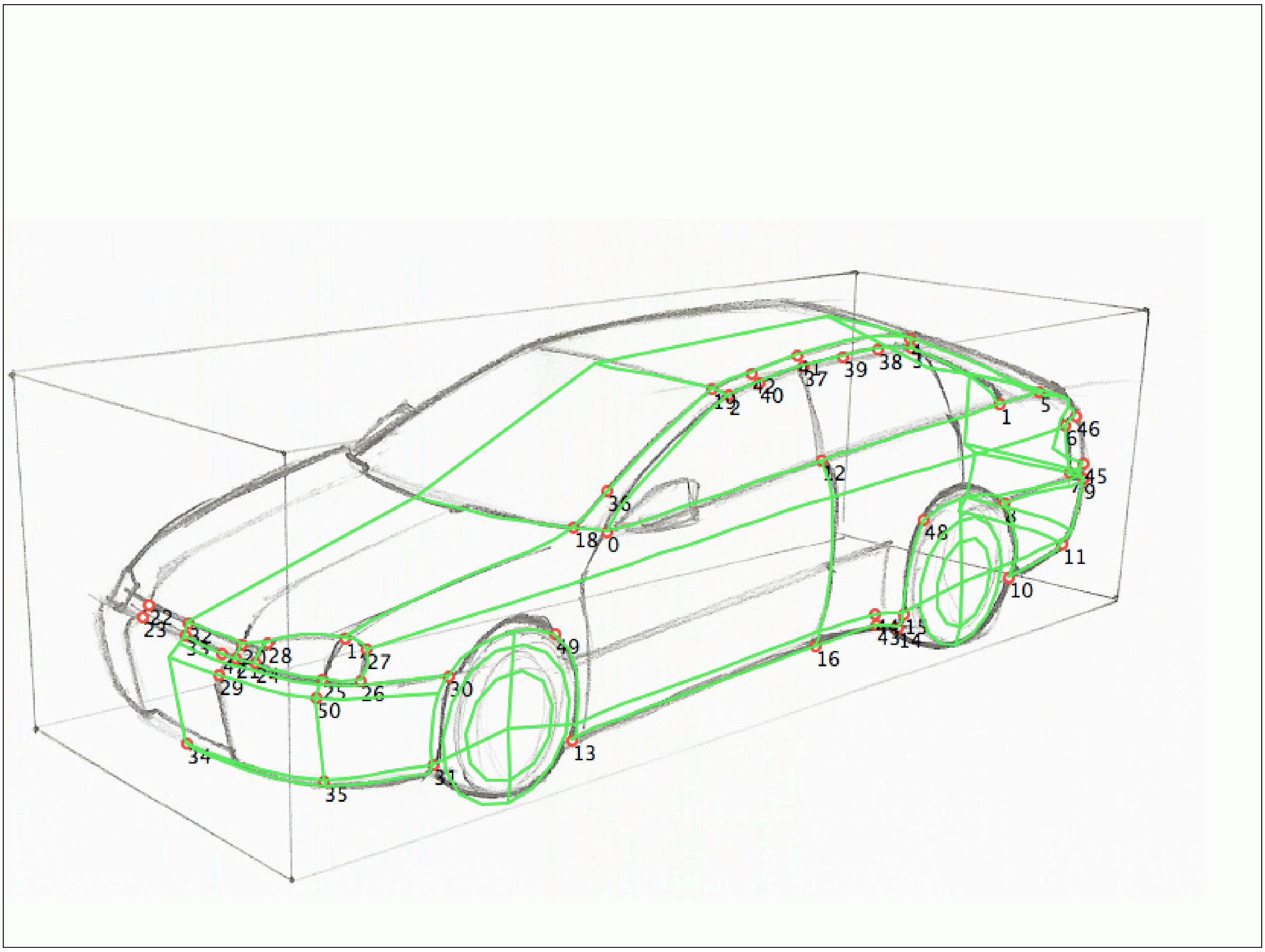
I’ll just want to show you one more demonstration to illustrate the strength of the system...


grill
headlight

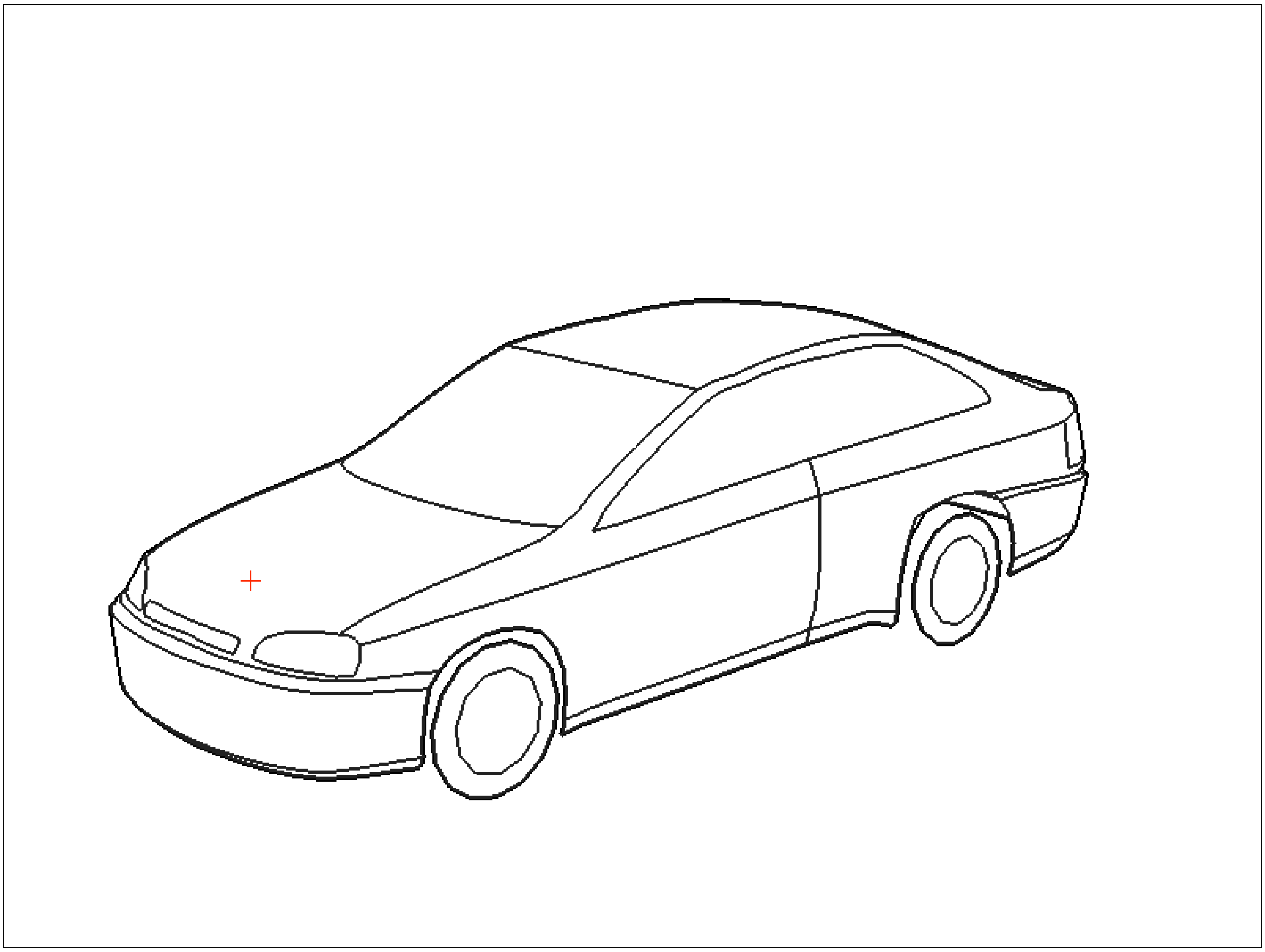


51

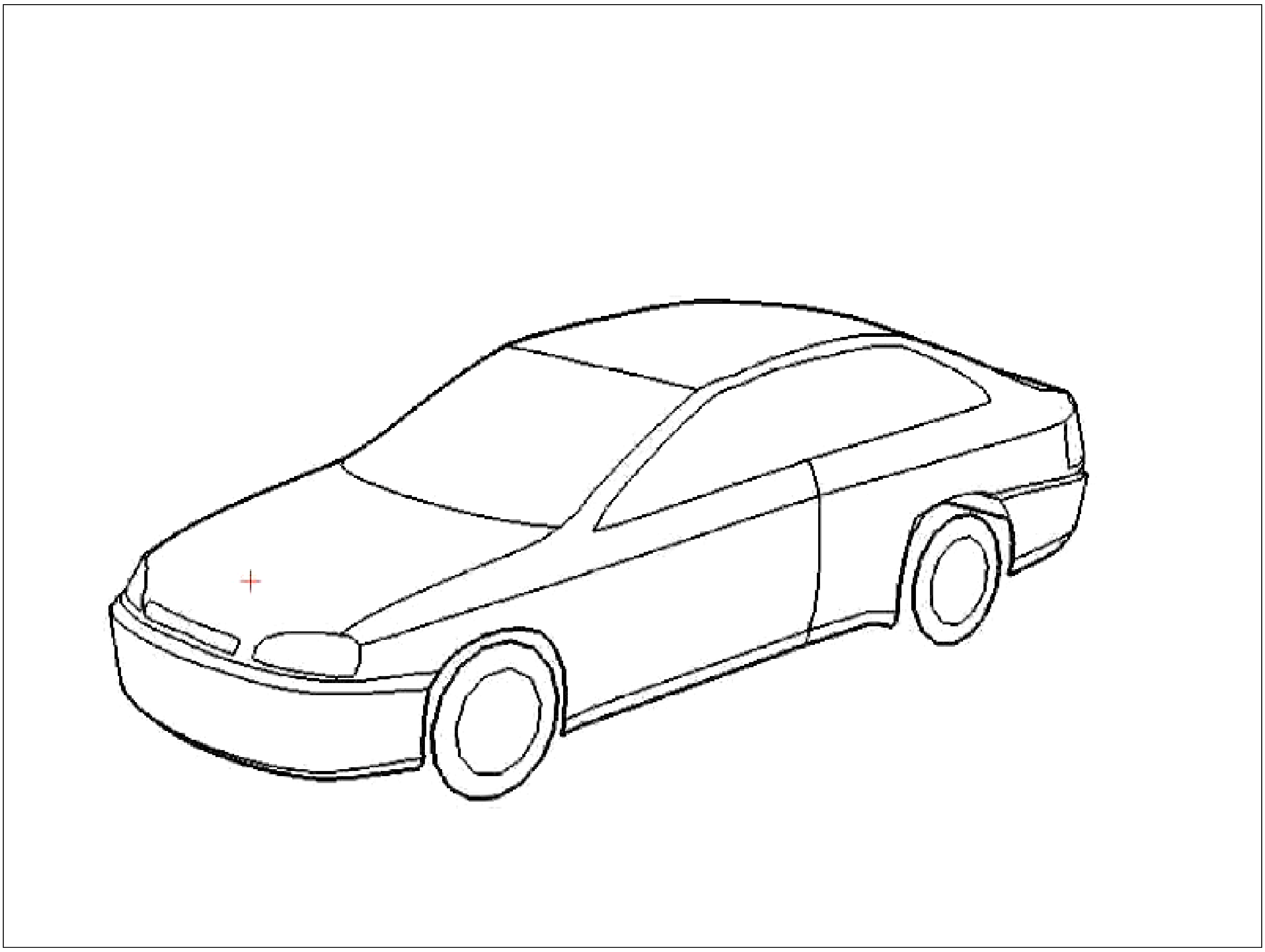
In this example we took a sketch where we know the projection matrix (can be calculated from the drawn bounding box). And marked manually some points on the sketch...



Here you can see the marked points (some fifty of them). We reconstructed the 3d shape from those points and you can see the green lines are rendered into the sketch. but I’ll show the 3d view also separately...



As you can see it looks nice, and last I’ll ...



show a short video for other viewpoints.

...

Limitations

- small set of examples
- fixed set of lines
- no 3d surface

http://informatik.unibas.ch/personen/istvan_kokai/sbim07/

Thank you very much, for more information, and additional materials visit the following site.

Now I'll happy to take your questions.