

QUO VADIS DESIGN

GENERATIVE ADVERSARIAL NETWORKS

ENTWERFEN 6. SEM. BACHELOR INNENARCHITEKTUR SOSE 2023

FAKULTÄT FÜR INNENARCHITEKTUR, ARCHITEKTUR UND DESIGN

PROF. ANETTE PONHOLZER

ENTWURF CLAUDIO SCHEMMERT

Kleines Lexikon der Fachbegriffe:

GAN - Generative Adversarial Network): ist eine Art von künstlicher Intelligenz, die aus zwei Komponenten besteht: dem Generator und dem Diskriminator. GANs werden verwendet, um Daten zu generieren, die ähnlich wie echte Daten aussehen.

NeRFs - Hyperparameter-Anpassungen: beinhaltet die Anpassung der Parameter des Modells, die seine Leistung und sein Verhalten beeinflussen, wie zum Beispiel die Lernrate, die Batch-Größe oder die Anzahl der Schichten in einem neuronalen Netzwerk.

Neuronales Netzwerk: ist ein Ansatz zur Modellierung von künstlicher Intelligenz, der von der Struktur und Funktionsweise des menschlichen Gehirns inspiriert ist. Es besteht aus einer Sammlung von miteinander verbundenen künstlichen Neuronen, die Informationen verarbeiten und lernen können.

Frontend: bezieht sich auf die Benutzeroberfläche (UI) einer Softwareanwendung oder einer Website. Es ist der Teil einer Anwendung, mit dem Benutzer interagieren und Informationen eingeben können.

Generator: erzeugt neue Daten, indem zufällige Eingaben erstellt und in einen datenähnlichen Zustand umgewandelt werden.

Discriminator: versucht, zwischen den vom Generator erzeugten Daten und echten Daten zu unterscheiden.

Epochen-Training: während einer Epoche werden die Trainingsdaten in kleinen Chargen (auch Batches genannt) dem neuronalen Netzwerk zugeführt. Das Netzwerk verarbeitet jedes Batch nacheinander und passt dabei seine Gewichte und Bias-Werte an, um die Leistung zu verbessern und das gewünschte Modellverhalten zu erlernen.

Random Noise: wird in der künstlichen Intelligenz (KI) oft verwendet, um eine gewisse Unsicherheit oder Variation in den Daten oder Modellen einzuführen. Es handelt sich um zufällige, nicht deterministische Signale, die den Daten hinzugefügt werden.

Training set: bezieht sich auf den Datensatz, der verwendet wird, um ein KI-Modell zu trainieren. Es besteht aus einer Sammlung von Beispieldaten, die als Eingabe für das Modell dienen, sowie den zugehörigen Zielwerten oder Labels, die das gewünschte Verhalten des Modells repräsentieren.

hidden layers: speziell in neuronalen Netzwerken beziehen sich "Hidden Layers" (versteckte Schichten) auf die Schichten zwischen der Eingabeschicht und der Ausgabeschicht. Diese Schichten werden als "versteckt" bezeichnet, weil sie nicht direkt mit den Eingabe- oder Ausgabewerten interagieren.

Input: bezieht sich auf die Daten oder Informationen, die einem KI-Modell zugeführt werden, um eine Vorhersage oder eine Verarbeitung durchzuführen. Der Input kann aus verschiedenen Arten von Daten bestehen, abhängig von der spezifischen Anwendung.

Output: bezieht sich auf das Ergebnis oder die Vorhersage, die von einem KI-Modell generiert wird, nachdem es den Input verarbeitet hat. Der Output kann je nach Art des KI-Modells und der spezifischen Aufgabe, die es löst, unterschiedliche Formen annehmen.

Batch-Size: kleinen Chargen (auch Batches genannt) werden dem neuronalen Netzwerk zugeführt.

Loss: nach jeder Epoche werden typischerweise Metriken wie der Verlust (Loss) und die Genauigkeit (Accuracy) des Modells berechnet, um den Fortschritt des Trainings zu bewerten

DESIGN- & ENTWICKLUNGSPROZESS

AUSGANGSSITUATION

Bei Projektstart von Quo Vadis Design war die Startidee aus Bildern und Texteingaben mit Hilfe eines KI-Programmes neue aufregende Designs zu entwickeln.

Im Zuge der Anfangsrecherche bin ich auf das Programm DreamFusion aufmerksam geworden. DreamFusion ist eine Weiterentwicklung von DreamFields und kombiniert Googles Open AI Bildmodellgenerator CLIP mit den 3D-Fähigkeiten von NeRFs.

Die Grundidee hierbei ist es 3-D Objekte aus 2-D Bildern zu generieren und über ein bereitgestelltes Neuronales Netzwerk zu erweitern, hierbei wird eine Veränderung durch Texteingabe erzeugt. Der Vorteil liegt hierbei darin, dass man keine Vorkenntnisse im Programmieren oder Entwerfen mit einem bestimmten Programm haben muss und so schnell und intuitiv zu einem Ergebnis kommt.

Bei der weiteren Recherche zu Neuronalen Netzwerken, GAN´s und der Frage wie funktioniert eine KI überhaupt kam es zu dem Entschluss das Thema bei 0 zu beginnen und aus dem Semester Projekt eine Art Selbstexperiment zu entwickeln.

Mit nur wenig Vorkenntnis im Programmieren, der Hilfe von Videos und Foren wurde das Ziel definiert eine KI selbst zu programmieren.

Die Grundidee war es eine KI mit Frontend zu entwickeln die Stühle generiert und dem Benutzer durch Eingabe der Anzahl von Stuhlbeinen, Material und Form verschiedene Bilder zur Ideen Entwicklung bereitstellt.

Nach dem Start und der näheren Beschäftigung mit dem Prozess wurde die Idee noch etwas vereinfacht, um eine Umsetzung innerhalb eines Semesters realistisch zu machen.

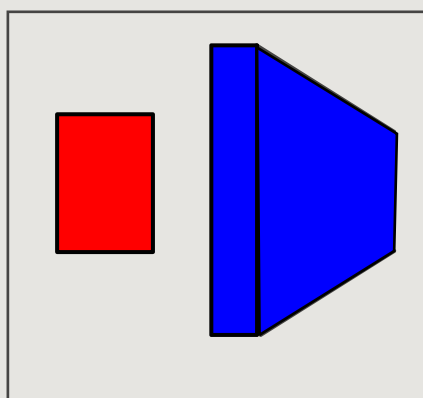
Ein selbst programmiertes Neuronales Netz aus Generator und Discriminator mit angehängter Datenbank wird trainiert und generiert am Ende des Epochen-Trainings eine Vielzahl verschiedener Stuhl-Bilder die als Grundlage für den weiteren Ideen und Designprozess fungieren.

WAS IST EIN GAN?

TRAINING SET

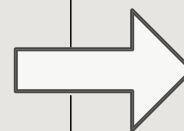


RANDOM NOISE



FAKE IMAGE

GENERATOR



DISCRIMINATOR

REAL
FAKE

HIDDEN LAYERS

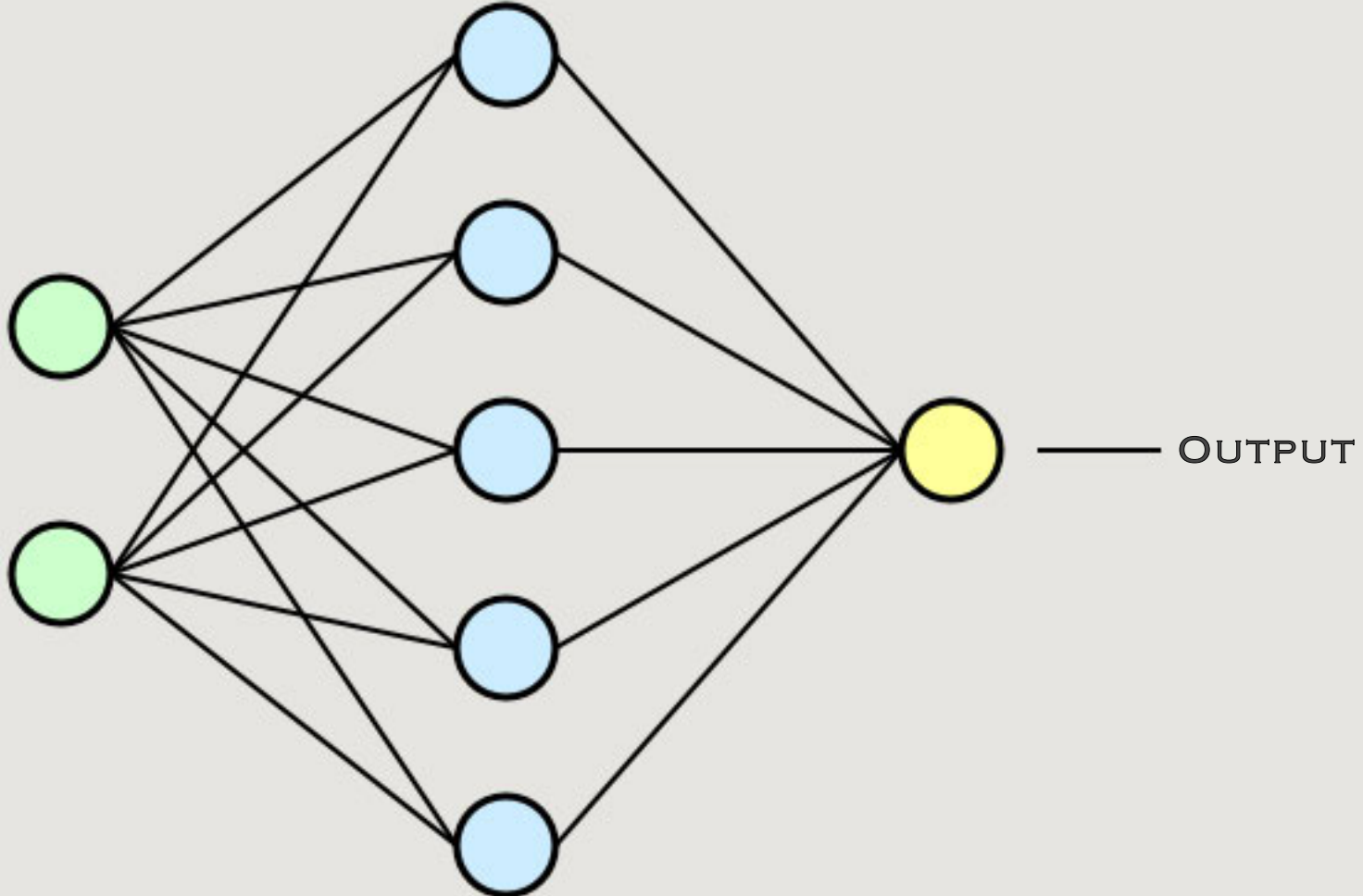
INPUT



INPUT



OUTPUT



INSIGHTS CODE

Der komplette Code hängt als .py-Datei an.

```
import os
print(os.listdir("/kaggle/input/stuhl-datenbank01"))
batch_size = 32
imageSize = 64
```

} Import der Datenbank
Festlegen der Batch-Size

```
random_transforms = [transforms.ColorJitter(), transforms.RandomRotation(degrees=20)]
transform = transforms.Compose([transforms.Resize(64),
                                transforms.CenterCrop(64),
                                transforms.RandomHorizontalFlip(p=0.5),
                                transforms.RandomApply(random_transforms, p=0.1),
                                transforms.ToTensor(),
                                transforms.Normalize((0.5, 0.5, 0.5), (0.5, 0.5, 0.5))
                            ])
```

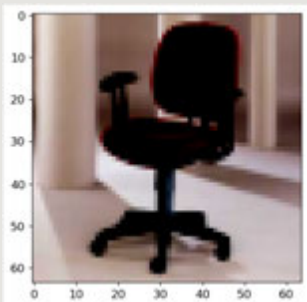
} Angleichen der Bilder
und Transformieren der
Datenbank

```
train_data = datasets.ImageFolder("/kaggle/input/stuhl-datenbank01", transform=transform)
train_loader = torch.utils.data.DataLoader(train_data, shuffle=True,
                                             batch_size=batch_size)
```

} Laden der angeglichenen
Bilder

```
imgs, label = next(iter(train_loader))
imgs = imgs.numpy().transpose(0, 2, 3, 1)
for i in range(2):
    plt.imshow(imgs[i])
    plt.show()
```

} Ausgabe zur Kontrolle



} Echter Stuhl aus der Datenbank

```
class Generator(nn.Module):
    def __init__(self, nz=128, channels=3):
        super(Generator, self).__init__()
```



Definieren des Generator
& Erweiterung um
Modulklassse „nn“

```
class Discriminator(nn.Module):
    def __init__(self, channels=3):
        super(Discriminator, self).__init__()
```



Definieren des
Discriminator

Hier verwendete Methode für effizienteres Lernen:

Init() Methode initialisiert die Parameter des Generators.

Die Parameter "nz" und "channels" geben die Größe des Latentvektors und die Anzahl der Kanäle (z.B. RGB-Farben) der generierten Bilder an.

Superklasse wird aufgerufen, um die init-Methode der Elternklasse nn.Module zu initialisieren. Auf diese Weise kann die Generator-Klasse Methoden und Attribute von nn.Module erben

```
# training:
batch_size = 32
LR_G = 0.001
LR_D = 0.0005

beta1 = 0.5

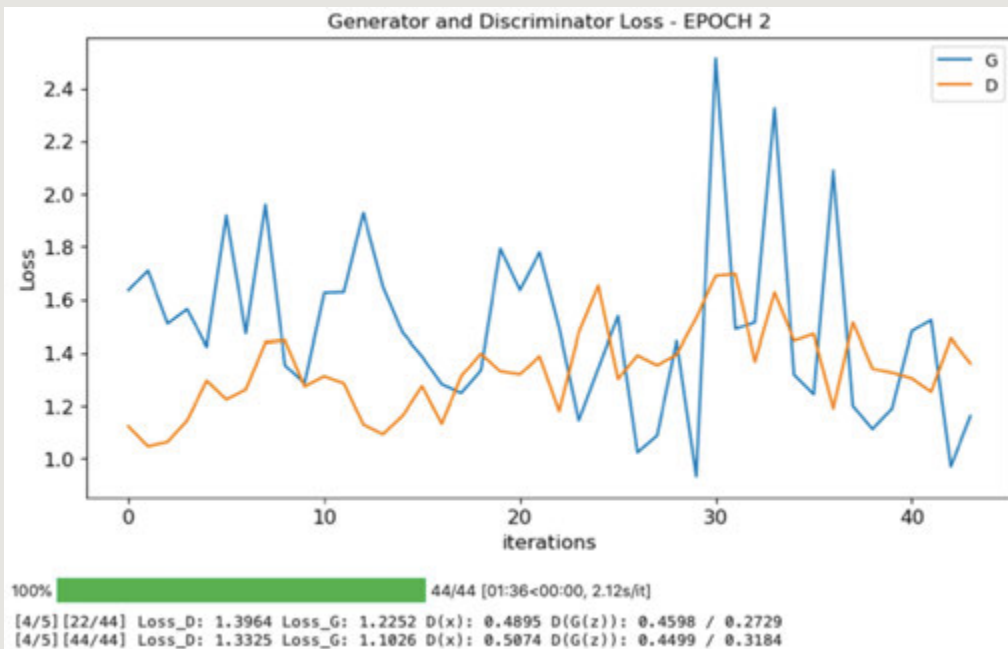
real_label = 0.9
fake_label = 0
nz = 128
```

```
# Epochen für Training
epochs = 5
```

```
show_generated_img()
```

} Festlegen der Batch-Size

} Festlegen der Lernrate
Angabe Anzahl Epochen
Ausgabe „KI“
generiertes Bild



} Nach jeder Epoche werden
Metriken wie Loss und Acc
festgelegt um den Fortschritt
des Trainings zu bewerten

ENTWURFSPROZESS

Vom KI - Pixel zum Stuhl

AUSWAHLPROZESS

Annäherung durch Skizzen

Nach Abschluss des Trainings und dem Generieren einer Vielzahl von Stühlen geht es nun an den Auswahl Prozess. Hier wurden aus der Sammlung die 15 Stühle mit dem höchsten Potenzial gewählt und skizzenhafte Ideen daraus entworfen.

Am Ende dieses Prozesses blieben drei Versionen und Skizzen mit interessantem und unkonventionellem Aussehen übrig

Ziel und Idee hierbei war es wieder einen Stuhl zu entwerfen der besonders ist und auch in Aussehen und Form einer KI basierten Inspirationsquelle gerecht wird.

Anbei werden die drei letzten und favorisierten Versionen und Skizzen gezeigt. Eine kritische Auseinandersetzung in Sachen Ergonomie und Aussehen ist hierbei erwünscht der Stuhl soll etwas darstellen, das nicht so leicht mit einem gewöhnlichen Designprozess entstehen kann.





VERSION 01
VERSION 01



VERSION 02



VERSION 03

DESIGNPROZESS

Ein Kunstobjekt Entwickelt aus der KI

Im Auswahlprozess wurde schnell klar, dass die Version 01 das interessanteste generierte Objekt ist. Die geschwungene Form und die vier etwas krummen und verschobenen Beine gaben eine gute Vorlage, um sich einen interessanten und nicht ganz gewöhnlichen Stuhl zu überlegen. Im 1. Schritt wurde die ursprüngliche Skizze überarbeitet und ergänzt, Ziel war es die ungewöhnlichen bunten Farbpunkte des KI-Bildes in einen baubaren Zustand zu bringen. Das sollte mit schief zu den Stuhlbeinen stehenden Stäben als Rückenlehne geschehen.

Im 2. Schritt rückte dann der Ursprungsgedanke des Experiments wieder in den Designprozess ein. Der Stuhl sollte von der Sitzschale her wie die KI generierte Vorlage geschwungen und elegant sein, ist aber von den Maßen so gewählt, dass ein bequemes Sitzen darin knapp nicht möglich ist. Zusätzlich und um den bunten „floralen“ Strukturen und Farben der Vorlage gerecht zu werden, wurde die Idee mit den Stäben wieder aufgegriffen.

Die durch die Sitzschale gesteckten Stäbe tragen zu dem Gefühl, das man sich gerne auf dem Stuhl setzen würde, aber eben nicht kann bei und runden das Objekt zusätzlich in Höhe und Aussehen ab.

Ein Kunstobjekt das sowohl zum Nachdenken über KI als auch zur Auseinandersetzung mit Design und Ergonomie einlädt.

Bezogen mit bunt gefärbtem Kunstfell und verarbeitet aus hochwertigen Materialien ist es letzten Endes das spannende Ergebnis der Zusammenarbeit zwischen Designer und Maschine.

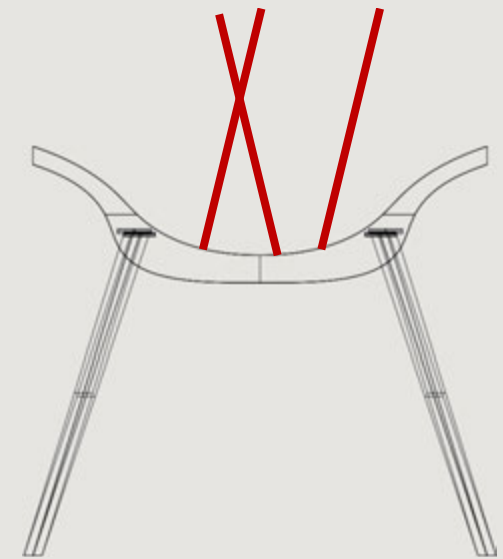
1. SCHRITT



GENERierterer STUHL



SKIZZE



CAD-ZEICHNUNG

2. SCHRITT



GENERIERTER STUHL



MODELL 3-D DRUCK



RENDERING



ENDMODELL

Rendering / Fotos / Technisches Zeichnungen

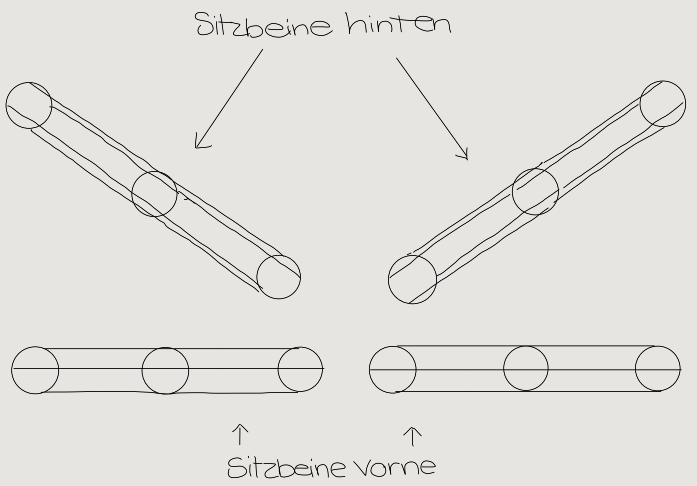
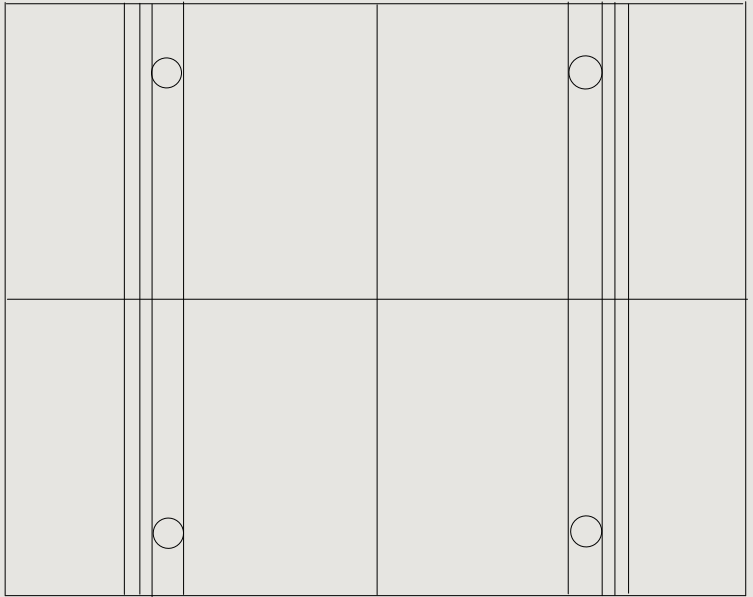
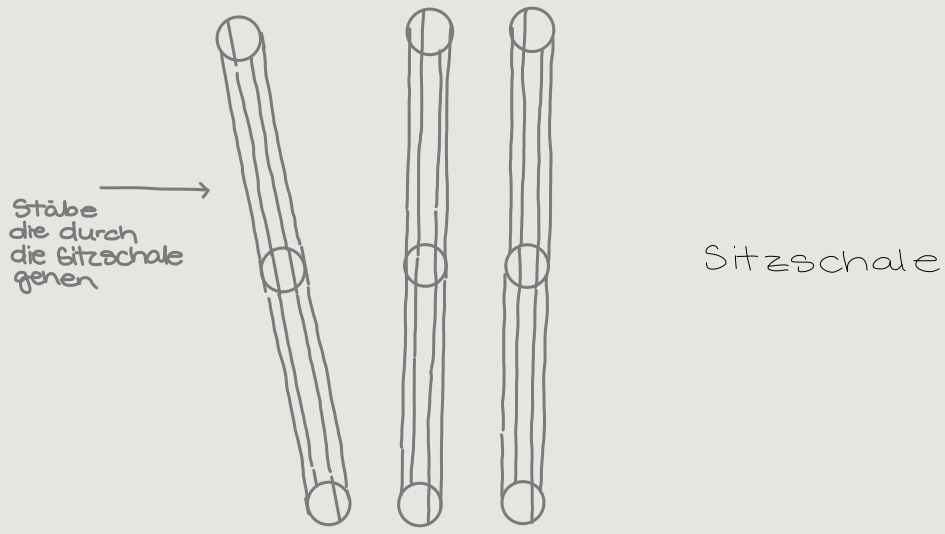




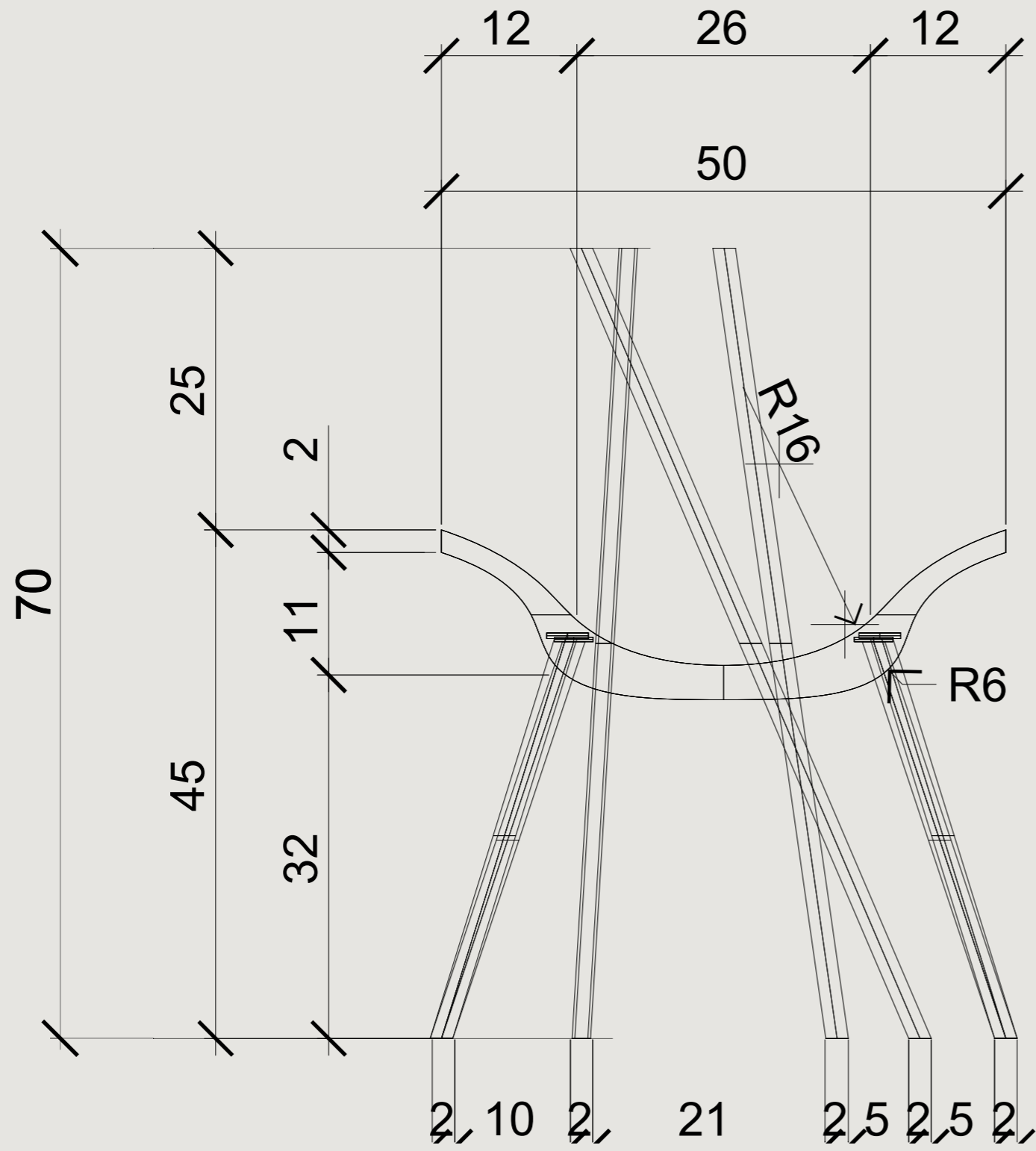




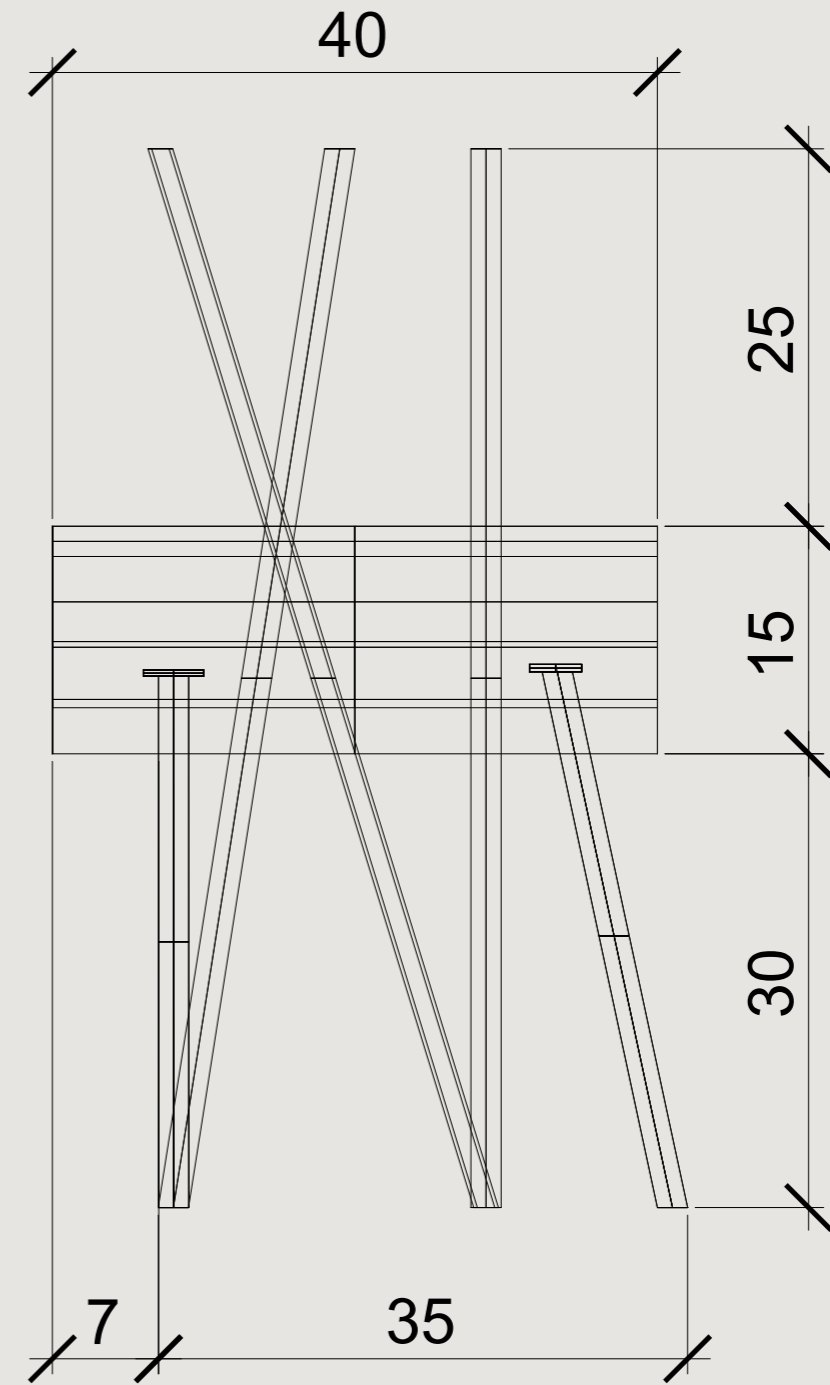
AUFBAU STUHL



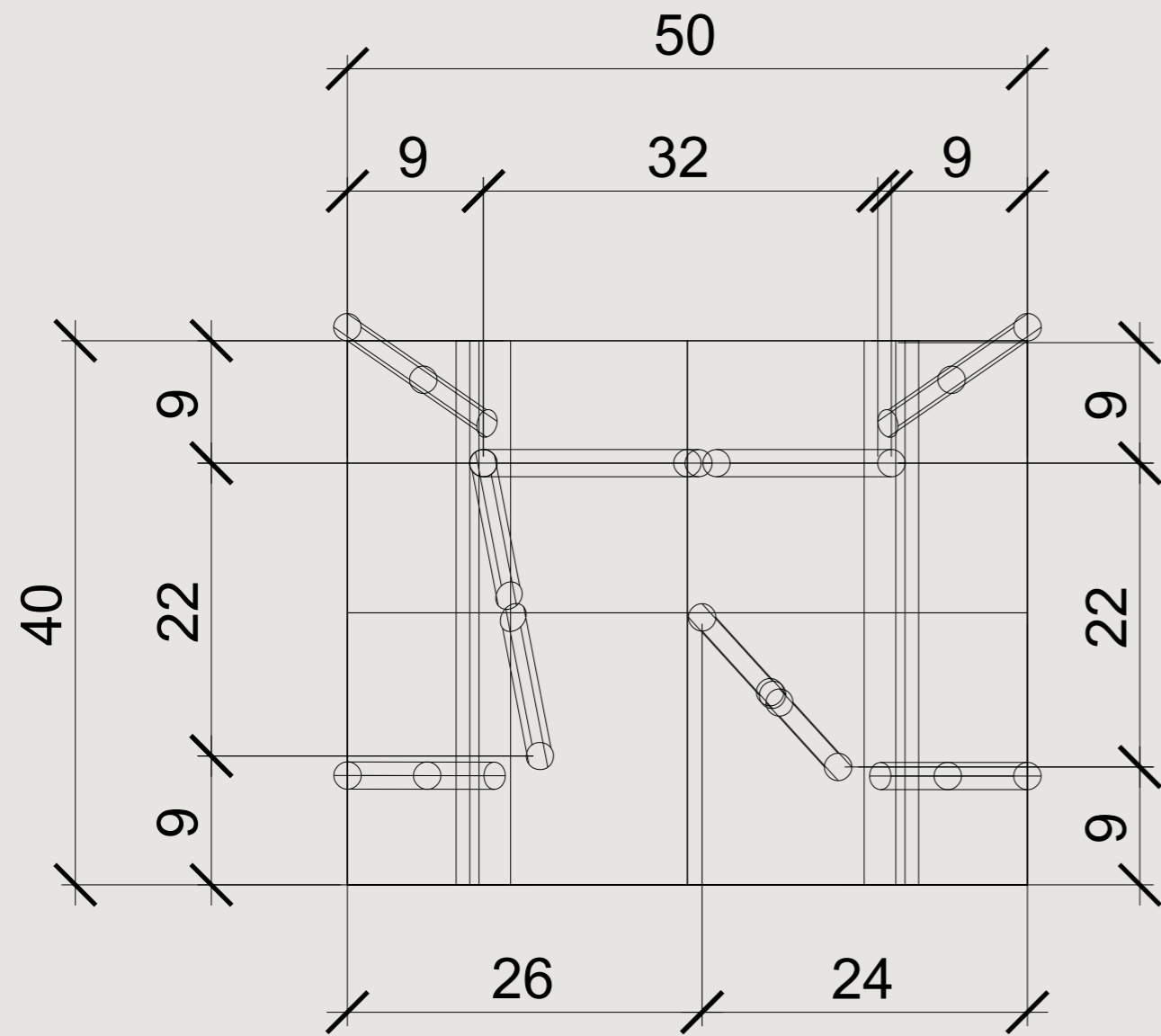
VORDER-ANSICHT M 1:5



SEITEN-ANSICHT M 1:5



DRAUFSICHT M 1:5



PROJEKTION EUROPÄISCH M 1:10

