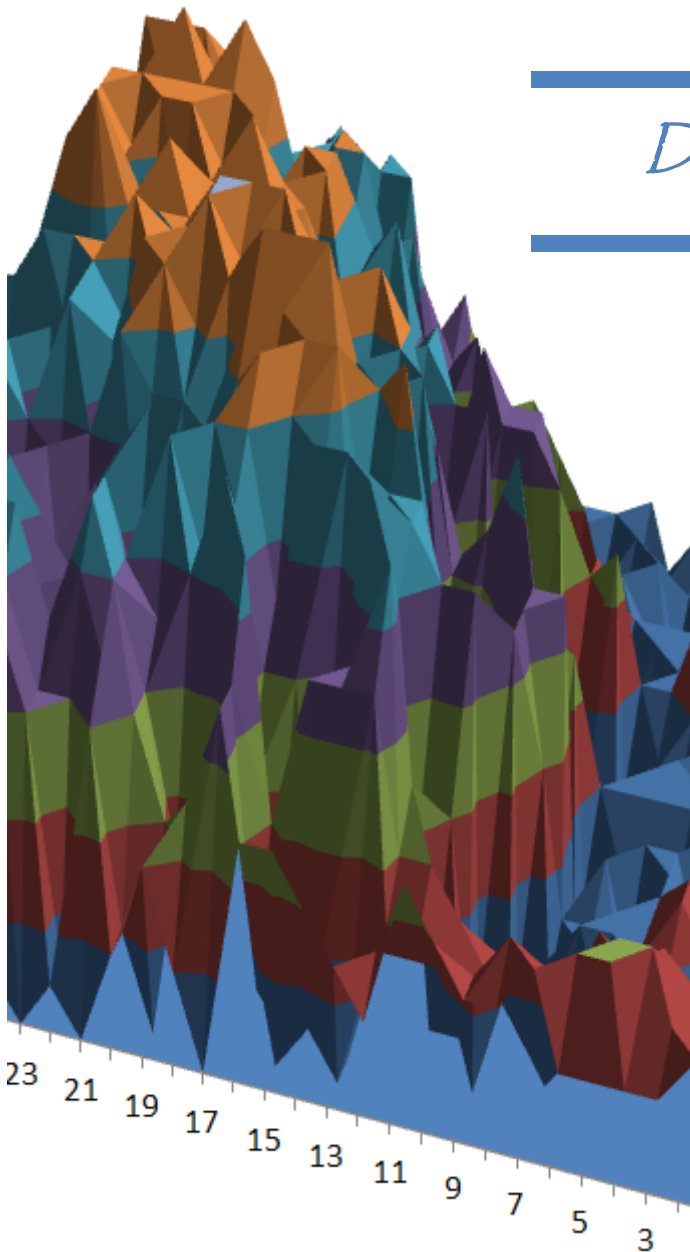


	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50		
9	0	4	5	2	0	1	2	1	0	2	0	1	0	0	0	1	3	4	2	3	8	8	7	8	8	8	6	7	7	6	4	1	0	0	0	2	0	0	0	0	0	3	6	4	0		
10	0	0	5	4	2	2	0	2	1	1	2	2	2	2	0	0	5	0	5	0	8	7	7	8	8	9	8	7	8	7	0	0	0	0	0	2	0	0	0	0	0	0	0	5			
11	5	4	5	3	1	1	0	1	1	2	1	2	1	0	4	3	4	4	0	7	9	8	8	10	9	8	8	7	7	0	0	0	0	0	0	0	0	0	1	1	0	0	5	5			
12	6	5	6	5	3	2	1	0	2	1	1	0	0	0	0	6	4	6	9	0	9	8	0	10	10	10	9	7	4	4	3	4	2	1	0	0	0	0	0	0	1	2	2	2	0		
13	7	5	5	4	2	3	2	0	1	2	2	0	4	3	3	7	2	9	9	8	9	9	9	10	11	11	11	7	4	3	5	0	0	0	0	0	0	1	0	3	0	2	0	1	2		
14	6	6	5	3	4	1	1	0	1	1	4	5	3	4	6	6	7	7	10	3	9	9	9	9	10	8	6	4	0	3	1	0	0	0	0	1	0	0	0	0	0	0	0	1			
15	1	3	6	5	5	4	2	2	2	1	4	5	6	5	7	6	8	8	10	10	9	11	12	11	12	10	10	10	1	3	6	1	0	1	0	2	2	0	1	0	1	2	1	0	1		
16	4	4	6	6	3	0	2	2	1	3	0	0	3	6	2	5	9	7	9	9	7	10	11	11	11	11	11	9	0	0	0	0	1	0	0	1	1	2	2	0	0	2	3	4	3		
17	6	6	6	4	2	3	2	0	0	4	0	6	4	7	3	7	7	9	9	9	9	11	10	11	11	10	11	10	5	6	8	9	9	10	1	0	1	2	1	0	2	0	2	0	3		
18	5	6	5	2	1	3	2	2	3	4	0	7	7	8	5	8	10	9	11	11	9	9	10	10	9	8	10	11	9	7	0	3	0	0	1	1	1	0	0	0	0	3	3	3	4		
19	6	4	3	3	3	2	1	0	5	1	6	7	7	5	4	8	8	9	11	8	10	9	10	11	10	11	10	9	7	0	5	4	3	0	0	0	0	0	0	3	2	2	1	3	2		
20	3	3	0	1	2	1	1	0	1	1	0	0	0	3	9	8	8	9	9	11	12	10	10	10	12	11	9	7	1	3	1	0	1	0	1	2	2	2	1	3	2	0	1	1			
21	3	2	2	1	1	2	1	0	0	1	4	3	8	8	9	7	8	9	11	11	11	11	10	10	9	7	8	6	5	7	1	1	2	1	2	2	1	2	2	1	0	0	0	1	1		
22	4	4	2	3	2	2	2	1	1	3	1	0	0	5	6	9	8	5	10	11	11	10	10	10	11	10	10	10	10	9	9	7	5	4	4	4	2	2	1	2	2	1	2	0	3	3	2
23	3	2	3	2	2	2	2	2	1	5	0	0	6	8	10	11	8	11	12	12	12	11	10	9	10	9	8	7	5	3	1	2	3	2	3	1	2	2	2	1	0	0	1	3	0		
24	2	4	3	2	1	1	1	0	5	0	0	7	6	1	10	9	11	11	10	11	12	10	11	9	10	9	8	7	1	1	1	2	2	3	3	1	3	2	1	2	1	2	1	2	0		
25	4	2	1	0	2	2	0	0	1	6	9	0	3	10	7	10	12	12	12	11	10	11	10	1	10	5	9	8	6	6	5	4	2	3	3	0	3	0	1	2	2	3	2	1	0		
26	4	3	3	1	2	0	2	0	7	7	0	0	7	0	10	9	10	11	11	0	9	9	10	8	7	8	9	9	6	7	4	5	3	5	1	2	1	0	0	0	2	1	2	1	0		
27	3	3	3	1	2	0	1	1	4	7	4	0	7	11	8	11	11	10	10	9	4	6	0	0	8	6	7	3	6	0	7	6	3	3	2	0	2	2	0	1	2	3	1	0	0		
28	2	3	1	3	2	2	1	0	0	7	0	0	9	9	7	9	10	11	6	8	5	9	8	0	7	8	7	5	5	4	7	4	1	2	3	2	2	1	1	2	3	0	0	0	0		
29	3	2	2	3	2	2	1	0	0	8	8	9	8	10	10	7	10	10	8	10	8	6	7	8	8	4	6	5	5	6	4	0	1	2	2	1	0	1	2	0	0	0	0	0	0		
30	2	1	1	2	2	1	0	0	1	0	4	3	8	10	0	6	6	9	8	10	7	9	7	8	9	8	0	6	5	1	4	3	2	3	3	3	2	0	2	2	1	0	0	0	0		
31	2	3	2	3	2	3	1	2	2	0	0	5	9	7	3	2	4	10	8	7	8	9	5	7	8	6	7	5	4	4	1	2	3	1	2	1	1	2	0	0	1	0	0	0	0		
32	2	2	3	3	1	0	1	2	3	4	4	7	7	7	4	9	3	3	6	8	8	7	9	9	8	5	6	5	2	0	1	0	1	1	1	1	1	2	0	1	0	0	0	0	0		
33	2	2	3	4	4	2	3	3	0	2	3	3	0	2	0	1	7	1	4	6	7	6	6	8	6	5	3	5	3	1	1	2	1	2	0	0	1	2	0	0	0	0	0	0	0		
34	3	2	3	3	4	4	3	4	3	3	3	3	0	1	0	3	7	5	0	2	6	1	6	8	6	5	1	0	0	1	2	1	2	1	0	0	2	2	0	0	0	0	0	2	0		
35	3	3	2	2	2	2	4	3	3	4	5	2	3	4	5	5	0	2	4	2	0	1	0	1	1	2	2	2	4	2	3	2	1	1	0	0	2	1	1	0	1	0	0	0	0		



Der große 3D-Scanner

Schüler experimentieren 2015

Mike Dederichs

St. Michael - Gymnasium

Bad Münstereifel

Der große 3D-Scanner

- 1 Kurzfassung**
- 2 Zielsetzungen**
- 3 Der Ultraschall-3D-Scanner**
 - 3.1 Versuchsplanung**
 - 3.2 Die Schaltung**
 - 3.3 Die Mechanik**
 - 3.4 Das Programm**
- 4 Auswertung**
- 5 Kritische Analyse des Scanergebnisses**
- 6 Danksagung**
- 7 Linkliste**

1 Kurzfassung

3D-Scanner können Objekte dreidimensional erfassen. Jedoch sind solche Scanner nicht nur teuer und komplex, sie können außerdem nur recht kleine Objekte einscannen. Mit meinem großen 3D-Scanner kann ich auch größere Objekte scannen. Dazu benutze ich die einfach programmierbare Arduino-Umgebung und insgesamt drei Ultraschallsensoren. Der Vorteil liegt so in einem erweiterbaren 3D-Scanner der insbesondere sehr kostengünstig ist.

2 Zielsetzung

Ziel ist es einen Scanner zu bauen, der kostengünstig auch große Objekte dreidimensional erfassen kann. Dazu sollen preiswerte Sensoren und ein günstiger Controller verwendet werden, der außerdem einfach zu programmieren ist. Die Sensoren erfassen die Lage im Raum und die Entfernung zum Objekt. So kann das Objekt dreidimensional erfasst werden.

3 Der Ultraschall-3D-Scanner

3.1 Versuchsplanung

Da ich mich entschieden habe einen Ultraschall-Sensor für die 3D-Erfassung zu verwenden, benötigte ich noch einen Mikrocontroller mit entsprechender Software, die den Sensor ansteuert und seine Daten auswerten kann. Die Wahl fiel schnell auf einen Arduino. Ein Arduino ist ein Mikrocontroller-Board das sich besonders einfach programmieren und komfortabel mit meinem Sensor verwenden lässt.

Dieses will ich zusammen mit dem Sensor auf einem beweglichen Gestell befestigen, um reproduzierbare Scans durchführen zu können. Meine Scan-Apparatur möchte ich einfach per Hand bedienen.

3.2 Die Schaltung

Wenn ich den Arduino mit dem Sensor verbinde sieht der Schaltplan wie folgt aus:

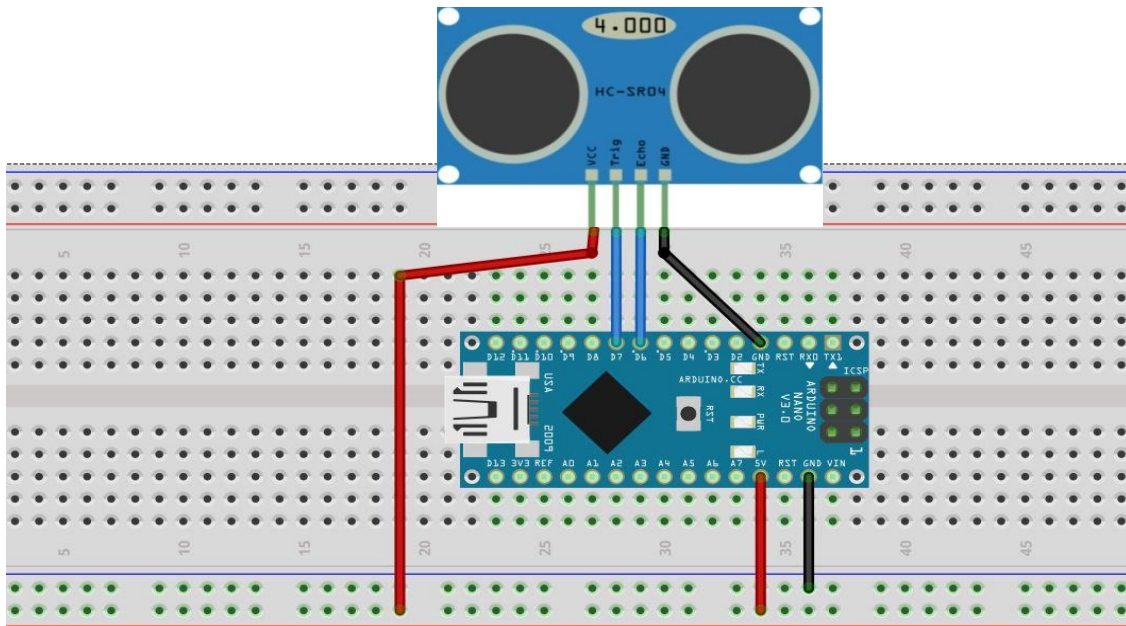


Abb. 3.1 Der Schaltplan - mit Arduino-Board und einem Ultraschallsensor

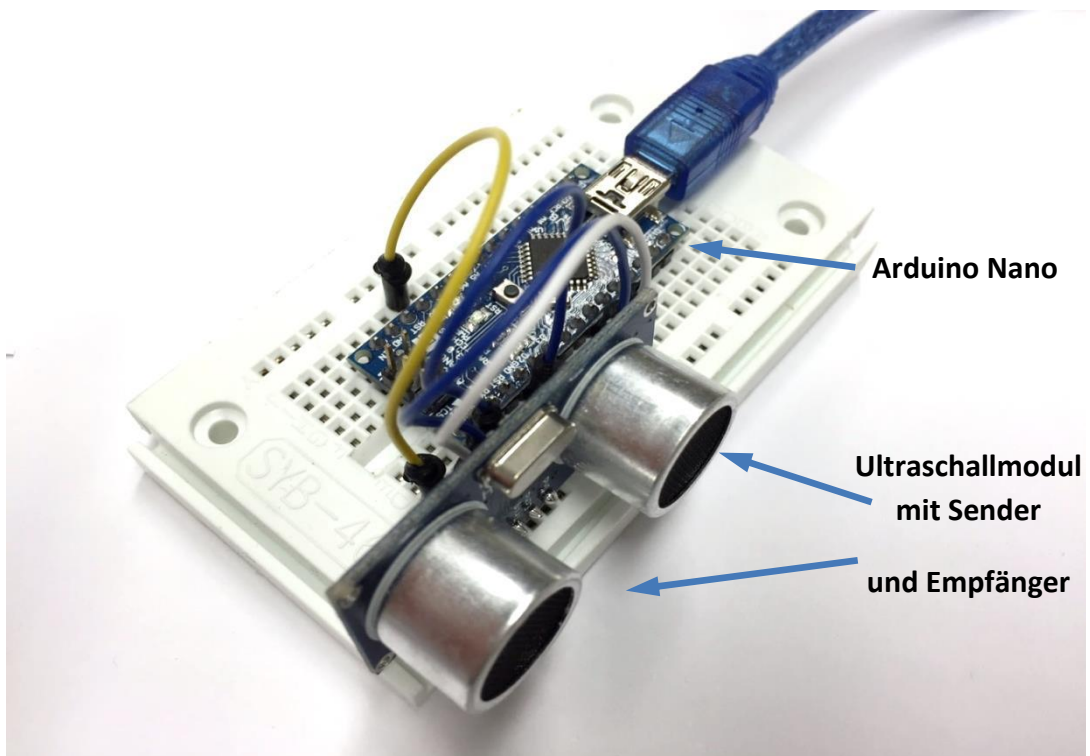


Abb. 3.2 Das Ultraschallmodul, angeschlossen an einen Arduino

Sowohl der Arduino als auch das Ultraschallmodul ist auf einem Steckbrett an eine Versorgungsspannung von 5V angeschlossen. Bevor ich den Arduino programmieren kann, muss ich zunächst verstehen wie ein Ultraschallmodul überhaupt funktioniert und wie man damit Entfernungen messen kann.

Das Ultraschallmodul besteht aus einem Sender und einem Empfänger. Um eine Entfernung zu bestimmen legt man einen kurzen Rechteckimpuls von 10 µs auf den Sender. Der Pin des Senders ist mit ‚Trig‘ für „Trigger“ beschriftet. Dadurch wird ein kurzes Ultraschallsignal erzeugt. Wird es von einem Objekt reflektiert, kann man den Impuls am Empfänger (am „Echo“-Pin) wieder messen. Aus der Laufzeit des Signals lässt sich mit der Schallgeschwindigkeit in Luft die Entfernung berechnen:

$$s = v \cdot t$$

$$\text{Strecke} = \text{Geschwindigkeit} \cdot \text{Zeit}$$

Die Schallgeschwindigkeit in Luft bei 20°C beträgt:

$$v = 343 \frac{m}{s}$$

Die Zeit messe ich mit Hilfe meines Moduls und kann dann die Strecke berechnen. Hier eine Beispielmessung (theoretische Werte).

Ich messe eine Signallaufzeit von **1470,6 µs = 0,0014706 s**.

Diesen Wert setzen wir in unsere Formel ein:

$$s = v \cdot t = 343 \frac{m}{s} \cdot 0,0014706 s \simeq 0,50 m = 50 cm$$

Jetzt muss ich aber daran denken, dass das Signal vom Modul abgesendet, reflektiert und wieder zurück läuft. Die gemessene Strecke beträgt also die doppelte Entfernung vom Objekt. Die Entfernung ist somit:

$$\text{Entfernung} = \frac{s}{2} = \frac{50 cm}{2} = 25 cm$$

Danach werden die Daten an den Computer übertragen. So ist es dann möglich die Daten zu sehen.

3.3 Die Mechanik

Als anspruchsvolles Objekt für die ersten 3D-Scans habe ich eine Landschaft aus Pappmaschee erstellt. Diese kann ich dann als Scanvorlage verwenden. Aber dafür muss ich mir zunächst eine sinnvolle Scan-Apparatur überlegen.



Abb. 3.3 Eine Landschaft aus Pappmaschee entsteht

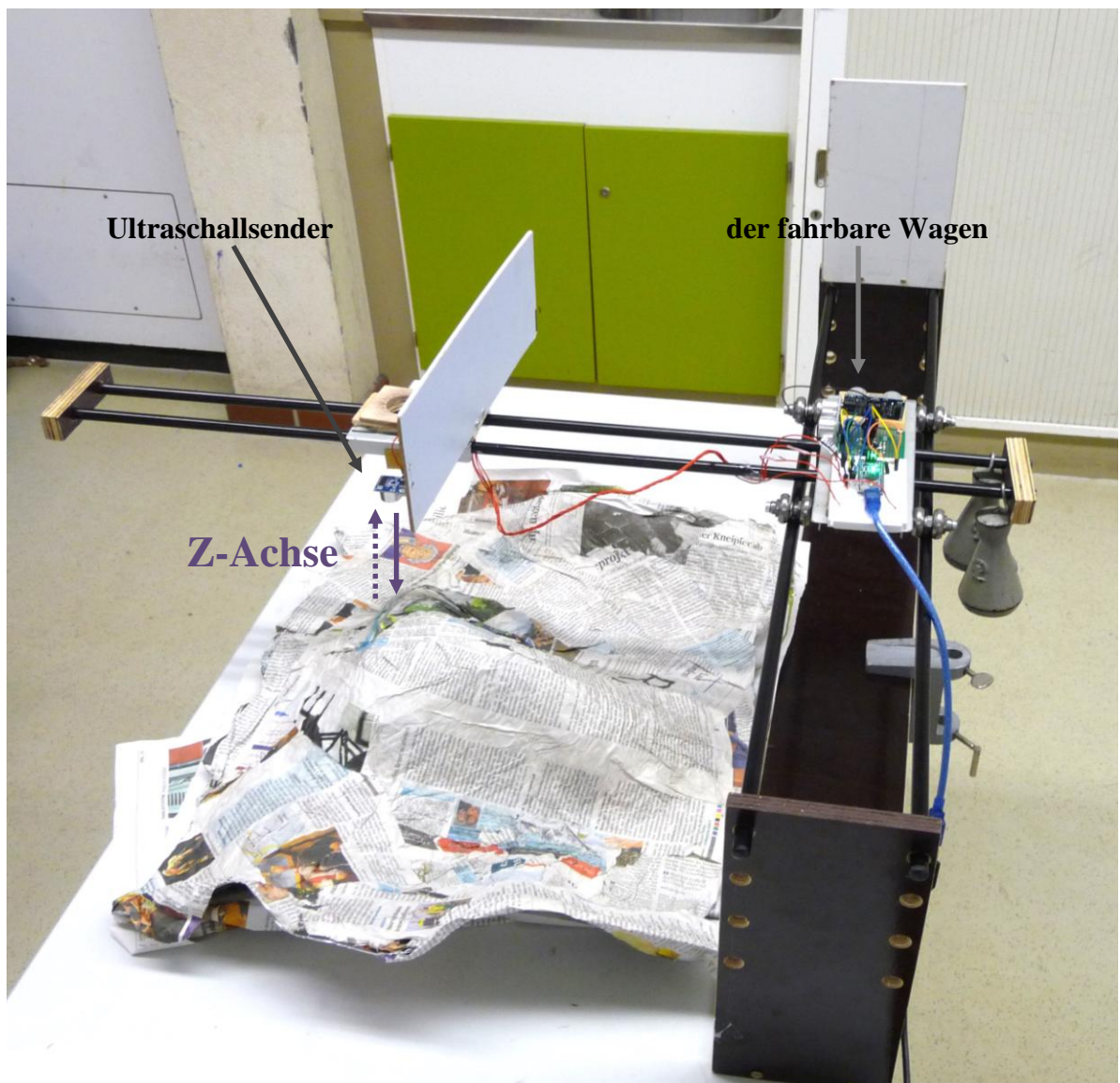


Abb. 3.4 Das Gestell meines 3D-Scanners

Ich habe ein Gestell gebaut um damit Messungen an großen Objekten durchführen zu können. Dafür habe ich zwei Bretter genommen und sie im rechten Winkel an die Enden eines langen Bretts gesetzt (Abb. 3.4). Seitlich habe ich Löcher in die Bretter gebohrt, um so Führungsschienen aus Kunststoff zu befestigen. Darauf lasse ich einen Wagen fahren, der auf Kugellagerrädern über die Stangen rollen kann (Abb. 3.5). An diesem Wagen sind ebenfalls Stangen angebracht um einen Ausleger zu befestigen. Mit insgesamt drei Ultraschallsensoren kann ich so die Position des Wagens und die Höhe über dem Objekt bestimmen, dass ich vermessen möchte. Zwei der Sensoren sind zusammen mit dem Arduino auf dem fahrbaren Wagen befestigt und messen die Position in X- und Y-Richtung. Am Ausleger ist der dritte Sensor angebracht. Dieser misst die Entfernung zu dem Objekt (Y-Richtung). Zusätzlich habe ich an dem Gestell weiße Holzbretter als Reflexionsflächen für die Ultraschallsensoren angebracht.

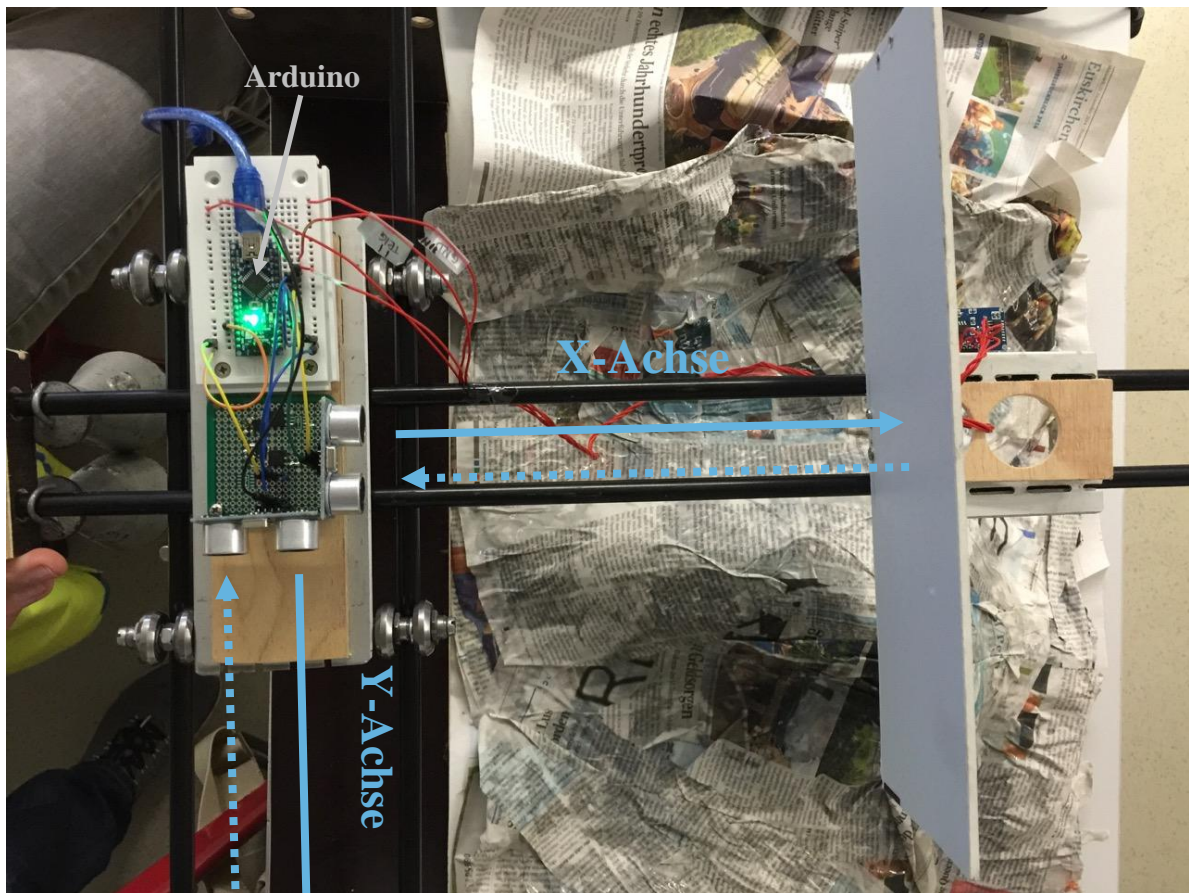


Abb. 3.5 Ansicht von oben: Der Wagen rollt auf den Führungsstäben
Zwei Ultraschallsensoren bestimmen die X- und Y-Position

3.4 Das Programm

Das Arduinoprogramm nutzt die Ultraschallmodule um den Abstand zum nächsten Objekt festzustellen. Dazu gibt das Ultraschallmodul für 20 Mikrosekunden einen Puls aus. Anschließend wird gemessen wie lange es dauert bis der Puls reflektiert und wieder empfangen wird. Im Programm sieht das ganze so aus:

Der Pin des Arduinos wird auf HIGH gesetzt, also angeschaltet, und der HIGH-Puls für 20 Mikrosekunden gesendet. Der Pin wird anschließend auf Low gesetzt, um zu messen, wann der Puls wieder zurückkommt.

`trigPin_z` ist hier der Pin der Arduinos, an dem der Trigger-Pin des Ultraschall-Moduls der Z-Richtung angeschlossen ist. In dem Beispiel oben wird also die Entfernung in Z-Richtung gemessen.


```
digitalWrite(trigPin_Z,HIGH);
delayMicroseconds(20);
digitalWrite(trigPin_Z,LOW);
Entfernung = pulseIn(echoPin_Z,HIGH)*0.01715;
```

Abb. 3.6 Der Programmcode der die Messung mit dem Ultraschallmodul ausführt

Dieser Programmteil wird nun auch für die X-, und Y-Richtung durchgeführt. Das ist der wichtigste Teil des Programms und wird in der Hauptschleife (Abb. 3.7) ausgeführt. Das restliche Programm sorgt für die Eingabe der zu Anfang gewünschten Parameter und natürlich für die Ausgabe auf dem Bildschirm.

```
while(1){
  nx,ny,nz = 0;
  Messung_laeuft = true;
  while(Messung_laeuft){
    if (nx < Anzahl_der_Messwerte ){
      Distanz_messen(X);
      delay(10);
      Distanz_messen(Y);
      delay(10);
      Distanz_messen(Z);
      delay(10);
    }

    Werte_ausgeben();

    delay(1000 / (Messungen_pro_Sekunde - 48) );
  }
}
```

Abb. 3.7 Die Hauptschleife des Programms

nx, ny und nz sagen aus, welche Messung in welcher Koordinatenrichtung in diesem Moment durchgeführt wird, also die 1. Messung, die 2. Messung, die 3. Messung - bis zur letzten Messung. Es werden so lange Messungen ausgeführt, bis diese Werte den zu Beginn des Programms festgelegten Wert `Anzahl_der_Messwerte` erreicht haben. Möchte man

zum Beispiel 100 Messungen in je X-, Y- und Z-Richtung haben, stellt man die Anzahl der Messwerte auf 100 ein.

Der darunterliegende Programmteil (in Abb. 3.7) wird mit `Messung_laeuft = true` aktiviert.

Nun wird die Distanz in allen drei Richtungen nacheinander gemessen. Die Funktion

`Distanz_messen(X)` ruft dabei genau den Programmteil von Abb. 3.6 auf, und nutzt in diesem Fall das Ultraschallmodul, das der X-Richtung zugewiesen wurde.

Anschließend werden die gemessenen Werte für alle drei Richtungen ausgegeben. Die entsprechende Funktion „`Werte_ausgeben()`“ ruft dabei diesen Programmteil auf:

Der `Serial.print`-Befehl gibt den Wert als Zahl auf dem Computerbildschirm aus. In Abb. 3.7

bleibt nun noch die letzte Zeile der Hauptprogrammschleife übrig: Hier wird dem Arduino

gesagt, wie lange es warten soll, bis die nächste Messung durchgeführt wird. Das wird ganz

einfach berechnet: 1000 ms geteilt durch die Anzahl der Messungen pro Sekunde die zu

Programmstart festgelegt wurden. Der Wert -48 entsteht, da die Zahl „0“ in der ASCII-

Tabelle an Stelle 48 liegt, also das ASCII-Zeichen 48 ist. Der Wert `Messungen_pro_Sekunde`

basiert auf dieser ASCII-Zeichenfolge und muss daher noch um diesen Wert angepasst werden.

```
void Werte_ausgeben(){
    static unsigned int k;
    for(k ; k < nz ; k++){
        Serial.print("\t");
        Serial.print(Array_mit_Sensormesswerten_X[k]);
        Serial.print("\t");
        Serial.print(Array_mit_Sensormesswerten_Y[k]);
        Serial.print("\t");
        Serial.println(Array_mit_Sensormesswerten_Z[k]);
    }
}
```

Abb. 3.8 Jeweils mit einem Tabstop getrennt (\t), wird der aktuelle Wert für X, Y und Z ausgegeben

```

turn)      45 2D 055 &#45; -      77 4D 115 &
          46 2E 056 &#46; .      78 4E 116 &
          47 2F 057 &#47; /      79 4F 117 &
scape)     48 30 060 &#48; 0      80 50 120 &
col 1)     49 31 061 &#49; 1      81 51 121 &
col 2)     50 32 062 &#50; 2      82 52 122 &
col 3)     51 33 063 &#51; 3      83 53 123 &

```

Abb. 3.9 Die Zahl 0 liegt an Stelle 48 der ASCII-Zeichenfolge

So sieht das Programm schließlich aus, wenn es ausgeführt wird und Daten auf dem Computer ausgibt (Abb. 3.10):

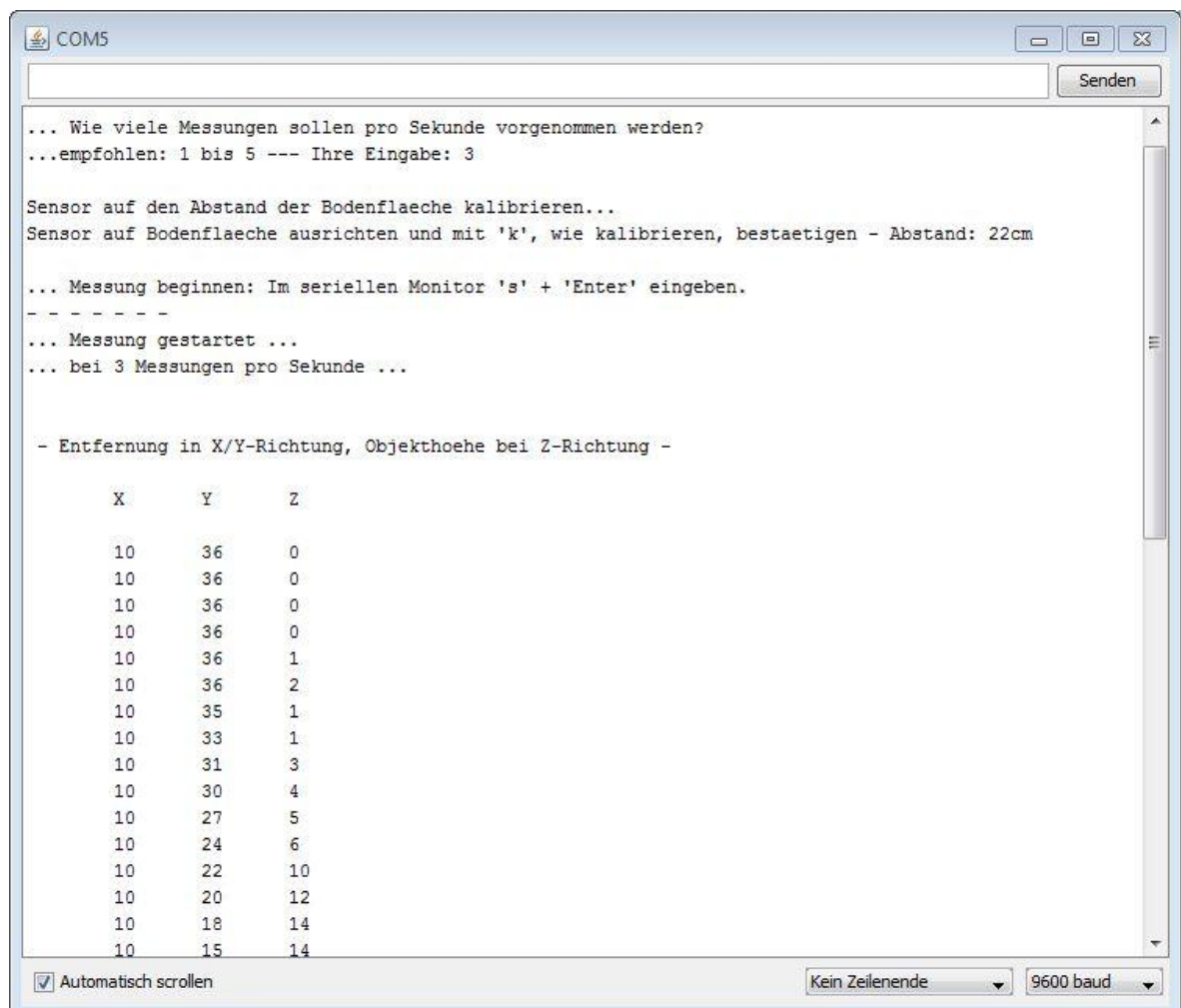


Abb. 3.10 Das Programm wird gerade ausgeführt und gibt die Messwerte für alle drei Richtungen als Entfernung in cm auf dem Bildschirm aus

4 Auswertung

Die Daten übertrage ich in Excel um damit ein Abbild meines Scans zu erhalten (Abb. 3.11).

	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
9	0	4	5	2	0	1	2	1	0	2	0	1	0	0	0	1	3	4	2	3	8	8	7	8	8	8	6	7	7	6	4	1	0	0	0	2	0	0	0	0	0	3	6	4	0
10	0	0	5	4	2	2	0	2	1	1	2	2	2	2	0	0	5	0	5	0	8	7	7	8	8	9	8	7	8	7	0	0	0	0	0	2	0	0	0	0	0	0	0	0	5
11	5	4	5	3	1	1	0	1	1	2	1	2	1	0	4	3	4	4	0	7	9	8	8	10	9	8	8	7	7	0	0	0	0	0	0	0	0	0	1	1	0	0	5	5	
12	6	5	6	5	3	2	1	0	2	1	1	0	0	0	0	6	4	6	9	0	9	8	0	10	10	10	9	7	4	4	3	4	2	1	0	0	0	0	0	0	1	2	2	2	0
13	7	5	5	4	2	3	2	0	1	2	2	0	4	3	3	7	2	9	9	8	9	9	9	10	11	11	11	7	4	3	5	0	0	0	0	0	0	1	0	3	0	2	0	1	2
14	6	6	5	3	4	1	1	0	1	1	1	4	5	3	4	6	6	7	7	10	3	9	9	9	10	10	8	6	4	0	3	1	0	0	0	0	1	0	0	0	0	0	0	1	
15	1	3	6	5	5	4	2	2	2	1	4	5	6	5	7	6	8	8	10	10	9	11	12	11	12	10	10	10	1	3	6	1	0	1	0	2	2	0	1	0	1	2	1	0	1
16	4	4	6	6	3	0	2	2	1	3	0	0	3	6	2	5	9	7	9	9	7	10	11	11	11	11	11	9	0	0	0	1	0	0	1	1	2	2	0	0	2	3	4	3	
17	6	6	6	4	2	3	2	0	0	4	0	6	4	7	3	7	7	9	9	9	9	11	10	11	11	10	11	10	5	6	8	9	9	10	1	0	1	2	1	0	2	0	2	0	3
18	5	6	5	2	1	3	2	2	3	4	0	7	7	8	5	8	10	9	11	11	9	10	10	9	8	10	11	9	7	0	3	0	0	1	1	1	0	0	0	0	3	3	3	4	
19	6	4	3	3	3	2	1	0	5	1	6	7	7	5	4	8	8	9	11	8	10	9	10	11	10	11	10	9	7	0	5	4	3	0	0	0	0	0	0	3	2	2	1	3	2
20	3	3	0	1	2	1	1	0	1	1	0	0	0	3	9	8	8	9	9	11	12	10	10	10	10	12	11	9	7	1	3	1	0	1	0	1	2	2	2	1	3	2	0	1	1
21	3	2	2	1	1	2	1	0	0	1	4	3	8	8	9	7	8	9	11	11	11	11	10	9	7	8	8	6	5	7	1	1	2	1	2	2	1	2	2	1	0	0	0	1	1
22	4	4	2	3	2	2	2	1	1	3	1	0	0	5	6	9	8	5	10	11	11	10	10	11	10	10	10	9	9	7	5	4	4	4	2	2	1	2	2	1	2	0	3	3	2
23	3	2	3	2	2	2	2	1	5	0	0	6	8	10	11	8	11	12	12	12	11	10	9	10	9	8	7	5	3	1	2	3	2	3	1	2	2	2	1	0	0	1	3	0	
24	2	4	3	2	1	1	1	0	5	0	0	7	6	1	10	9	11	11	10	11	12	10	11	9	10	9	8	7	1	1	1	2	2	3	3	1	3	2	1	2	1	2	1	2	0
25	4	2	1	0	2	2	0	0	1	6	9	0	3	10	7	10	12	12	12	11	10	11	10	1	10	5	9	8	6	6	5	4	2	3	3	0	3	0	1	2	2	3	2	1	0
26	4	3	3	1	2	0	2	0	7	7	0	0	7	0	10	9	10	11	11	0	9	9	10	8	7	8	9	9	6	7	4	5	3	5	1	2	1	0	0	0	2	1	2	1	0
27	3	3	3	1	2	0	1	1	4	7	4	0	7	11	8	11	11	10	10	9	4	6	0	0	8	6	7	3	6	0	7	6	3	3	2	0	2	2	0	1	2	3	1	0	0
28	2	3	1	3	2	2	1	0	0	7	0	0	9	9	7	9	10	11	6	8	5	9	8	0	7	8	7	5	5	4	7	4	1	2	3	2	2	1	1	2	3	0	0	0	0
29	3	2	2	3	2	2	2	1	0	0	8	8	9	8	10	10	7	10	10	8	10	8	6	7	8	8	4	6	5	5	6	4	0	1	2	2	2	1	0	1	2	0	0	0	0
30	2	1	1	2	2	1	0	0	1	0	4	3	8	10	0	6	6	9	8	10	7	9	7	8	9	8	0	6	5	1	4	3	2	3	3	3	2	0	2	2	1	0	0	0	0
31	2	3	2	3	2	3	1	2	2	0	5	9	7	3	2	4	10	8	7	8	9	5	7	8	6	7	5	4	4	1	2	3	1	2	1	1	2	0	0	1	0	0	0	0	
32	2	2	3	3	3	1	0	1	2	3	4	4	7	7	7	4	9	3	3	6	8	8	7	9	9	8	5	6	5	2	0	1	0	1	1	1	1	2	0	1	0	0	0	0	
33	2	2	3	4	4	2	3	3	0	2	3	3	0	2	0	1	7	1	4	6	7	6	6	8	6	5	3	5	3	1	1	2	1	2	0	0	1	2	0	0	0	0	0	0	
34	3	2	3	3	4	4	3	4	3	3	3	3	0	1	0	3	7	5	0	2	6	1	6	8	6	5	1	0	0	1	2	1	2	1	0	0	2	2	0	0	0	0	2	0	
35	3	3	2	2	2	2	4	3	3	4	5	2	3	4	5	5	0	2	4	2	0	1	0	1	1	2	2	2	4	2	3	2	1	1	0	0	2	1	1	0	1	0	0	0	0

Abb. 3.11 Die Höhenwerte in cm in einer Exceltabelle. Da die bedingte Formatierung aktiviert ist, erkennt man schon jetzt den Berg meiner Landschaft in

Aktiviert man in Excel die bedingte Formatierung von Zellen, werden die Zahlenwerte je nach Betrag farblich markiert. So kann man schon jetzt ein Höhenprofil erkennen. Große Zahlen werden grün markiert, niedrige Werte rot.

Nun kann man aus diesen Werten ein 3D-Diagramm erstellen. Das Ergebnis des 3D-Landschaftsscans sieht man in Abb. 3.12 .

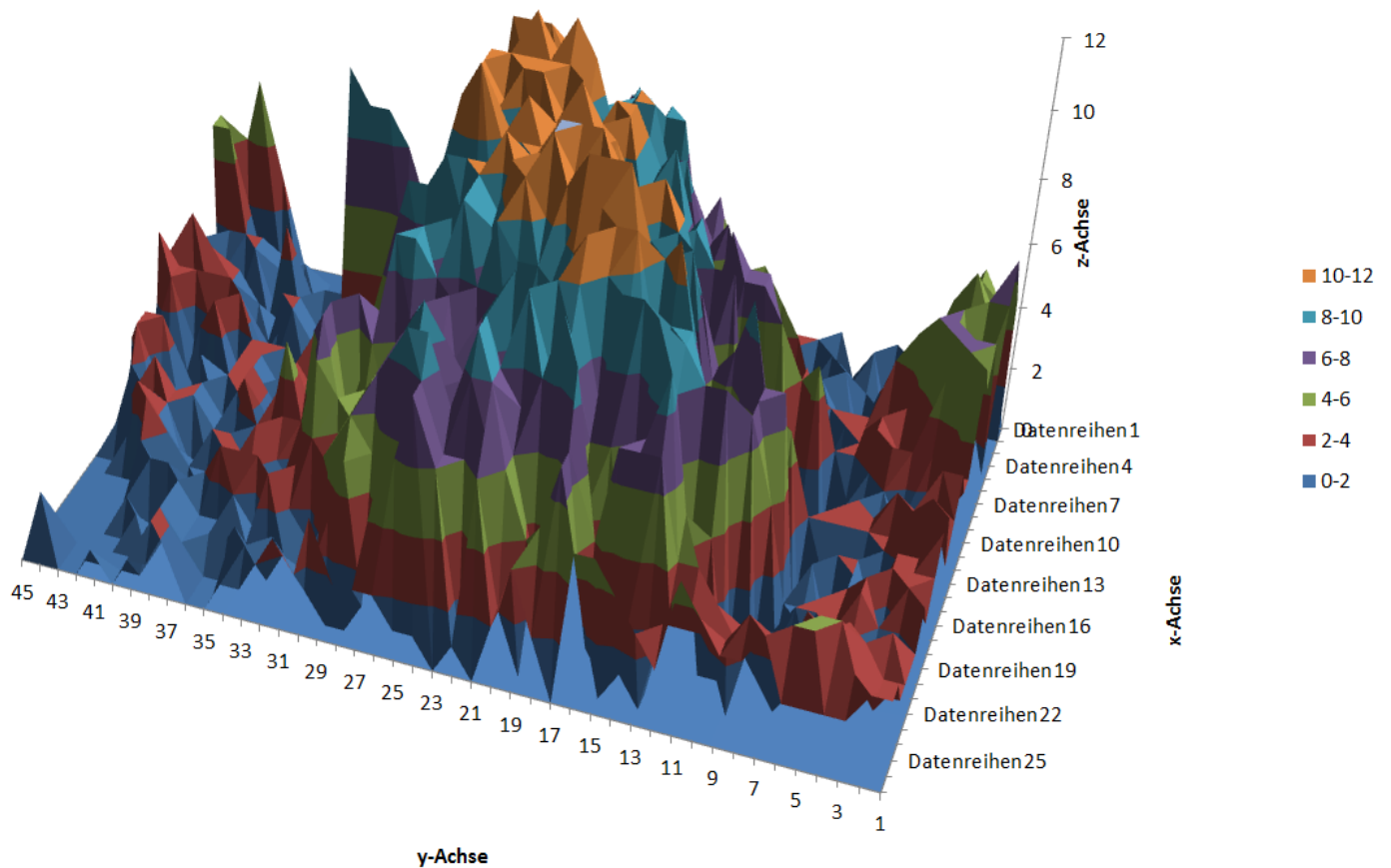


Abb. 3.12 Ein 3D Oberflächendiagramm der Exceltabelle

5 Kritische Analyse des Scanergebnisses

Mit dem Ultraschallmodul lässt sich das Höhenprofil schon gut erfassen. Dennoch treten einige Probleme mit dieser Methode auf: Ist das Profil zu steil, wird an diesen Bereichen der Schall nicht zum Empfänger zurückreflektiert. So wird an dieser Stelle der falsche oder kein Höhenwert ermittelt (Abb. 3.13).

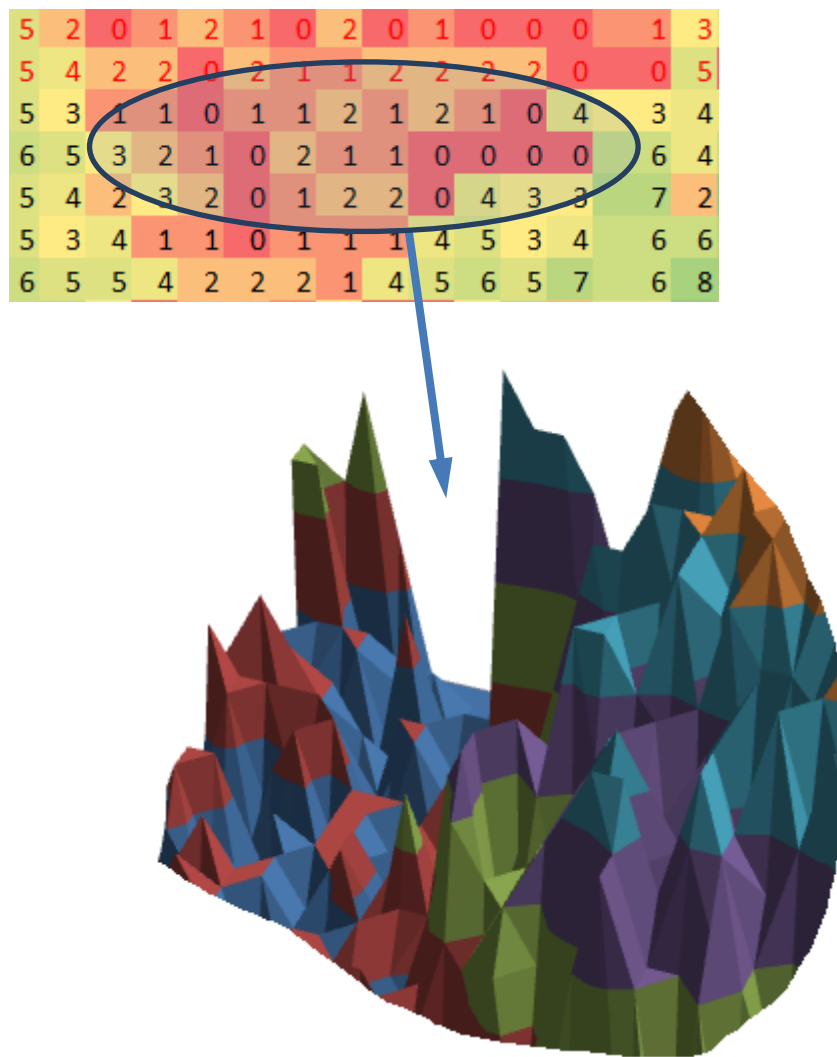


Abb. 3.13 Ein Fehler beim Scannen zu steiler Hänge. Der Schall wird hier nicht richtig korrekt reflektiert.

Dies ist auch dann problematisch, wenn bei unregelmäßig geformten Profilen einzelne Messwerte nicht erfasst werden und das Profil so eine Lücke hat (Abb. 3.14).

Abgesehen von den Problemen die durch die Reflexion des Ultraschalls entstehen ist die Auflösung der Sensoren begrenzt. Bei einer Entfernungsmessung wird eine Fläche von etwa 3cmx3cm erfasst. Dadurch entstehen die Spitzen im Diagramm. Excel bietet leider keine Funktion zum „runden Verbinden“ benachbarter Messwerte. So wäre es möglich die runden Flächen des Landschaftsprofils deutlich besser darzustellen.

11	10	11	12	10
12	12	11	10	11
11	11	0	9	9
10	10	9	4	6
11	6	8	5	9

Abb. 3.14

Der Wert „0“ dürfte hier nicht sein

6 Ausblick

Das Erfassen von großen Objekten erfüllt meine Erwartungen schon recht gut. Mein Gestell lässt sich mit dem aktuellen Aufbau auf ein Scannvolumen von 3m x 3m x 3m erweitern! Da die Messungen per Hand in Excel übertragen werden, ist es jedoch zeitaufwendig große Objekte zu scannen. Eine Erweiterung des Arduinoprogramms und eine automatisierte Messung würden mich bei diesem Prozess entlasten.

Vielversprechend sind erste Versuche, die Entfernungsbestimmung mit Ultraschall durch einen Laser zu ersetzen. Mit einem Laser lassen sich auch größere Entfernungen messen und so vielleicht sogar eine richtige Landschaft.

7 Danksagung

Vielen Dank an meinen Betreuungslehrer Walter Stein und auch an Herrn Stefan Hück, der mir bei Fragen zur Programmierung gute Tipps gegeben hat.

8 Linkliste

Arduino Webseite:	http://www.arduino.cc
Arduino & Ultraschall:	http://www.arduino-blog.de/2014/03/02/hc-sr04
Fritzing Schaltpläne erstellen:	http://www.fritzing.org