

Simulation of Heavy Rainfalls in Berlin

Sebastian Jäger



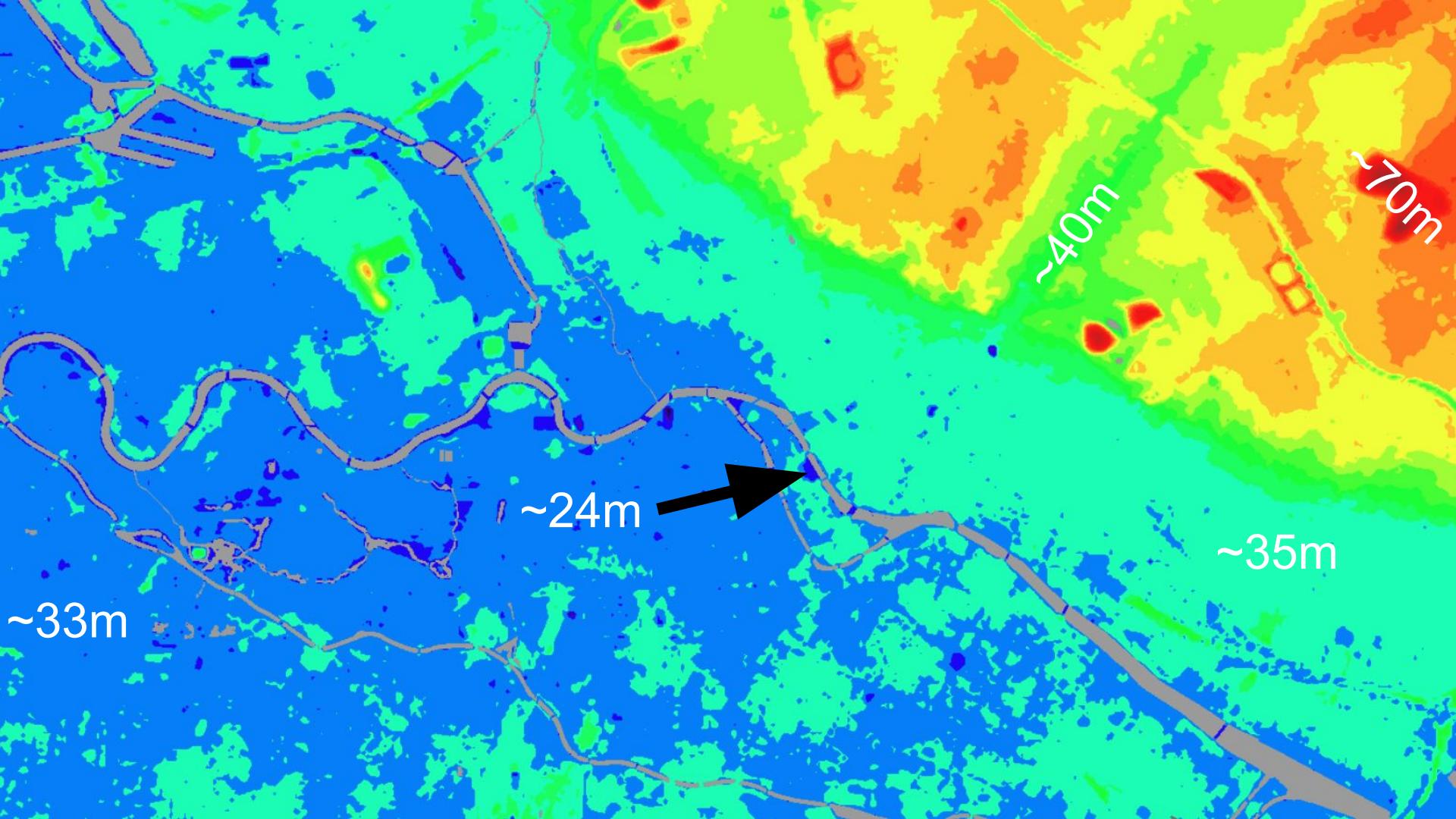
Customer:
Goal:
Data:

City of Berlin
Simulate dangerous areas

**BERLIN
OPEN
DATA**

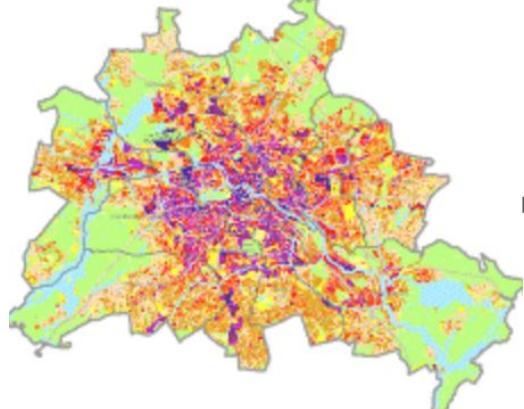
(daten.berlin.de)



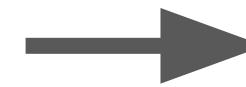


Sealing

Versiegelung 2016 (Ausgabe 2017)
Sealing 2016 (issue 2017)



QGIS



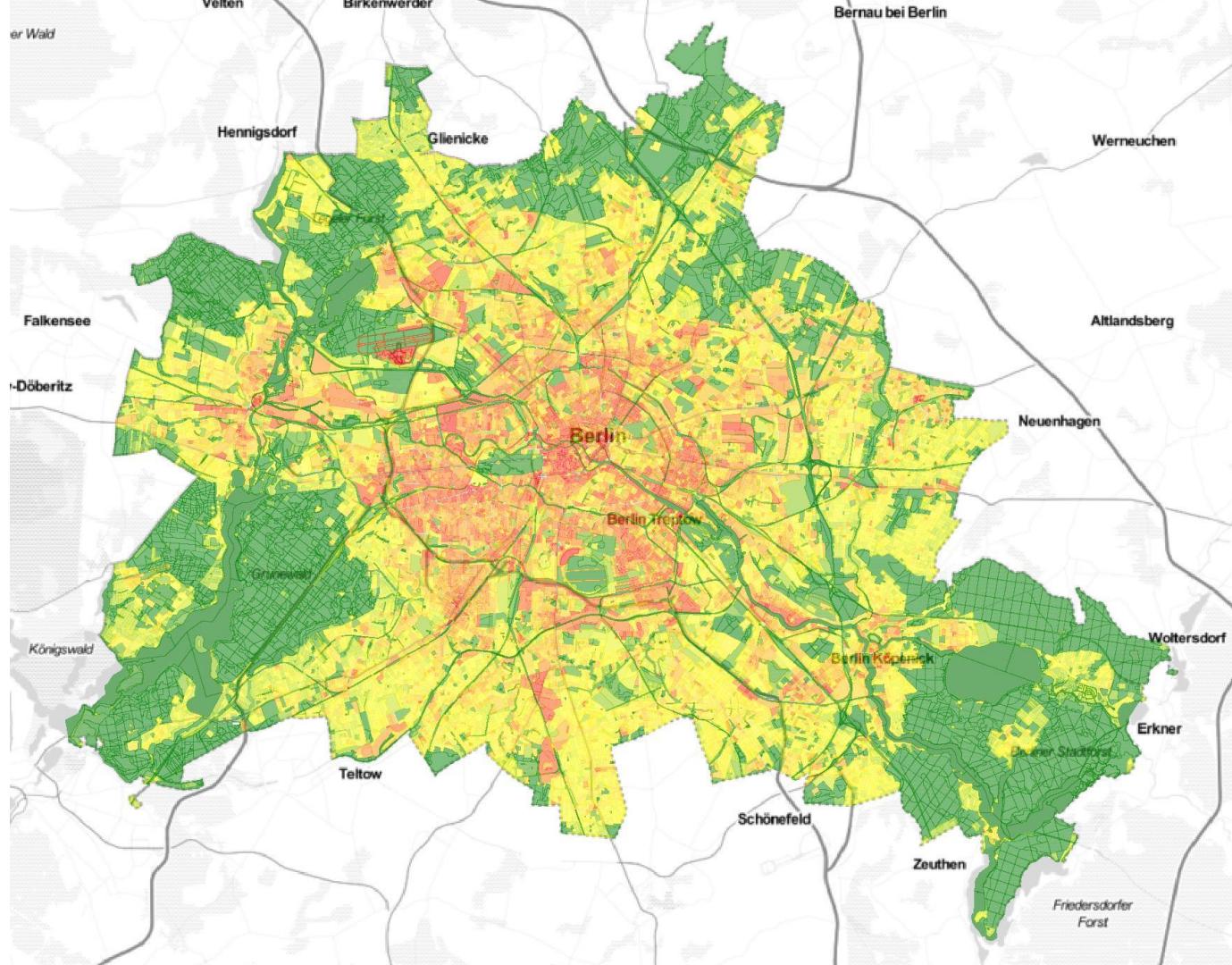
[Web Feature Service \(WFS\)](#)



**GeoPandas +
Feature Selection**



	sealing	geometry
0	69.30	POLYGON ((389326.214 5821756.291, 389374.948 5...
1	38.56	POLYGON ((389261.965 5821859.562, 389326.214 5...
2	83.53	POLYGON ((389261.965 5821859.562, 389326.319 5...
3	73.07	POLYGON ((389382.276 5821858.150, 389605.756 5...
4	76.37	POLYGON ((389382.276 5821858.150, 389488.086 5...





Ground Level

Digitale Geländemodelle – ATKIS DGM
Digital terrain models - ATKIS DGM

Granularity

ATKIS® DGM (1m-Rasterweite)

Kurzbeschreibung: Das ATKIS DGM ist ein Teil des Amtlichen Topographisch-Kartographischen Informationssystem (ATKIS).

Koordinatensysteme: EPSG:25833

Grafikformate: text/csv

Nutzungsbedingungen: Nutzungsbedingungen: Für die Nutzung der Daten ist die Datenlizenz Deutschland - Namensnennung - Version 2.0 anzuwenden. Die Lizenz ist über <https://www.govdata.de/dl-de/by-2-0> abrufbar. Der Quellenvermerk gemäß (2) der Lizenz lautet "Geoportal Berlin / [Titel des Datensatzes]".

Zugriffsbeschränkungen: Es gelten keine Bedingungen

Technische Angaben

ATOM-Feed-Url: https://fbinter.stadt-berlin.de/fb/feed/senstadt/a_dgm

Standard: ATOM-Feed

Version: 1.0

Direkter Download des Datensatzes

ATKIS® DGM (1m-Rasterweite) - Download-Service - EPSG25833 -  gezippt.

Erläuterung Blattschnittübersicht 2x2 Km

ATKIS® Basis-DGM1 370_5806 http://fbinter.stadt-berlin.de/fb/atom/DGM1/370_5806.zip

ATKIS® Basis-DGM1 370_5808 http://fbinter.stadt-berlin.de/fb/atom/DGM1/370_5808.zip

ATKIS® Basis-DGM1 370_5810 http://fbinter.stadt-berlin.de/fb/atom/DGM1/370_5810.zip

ATKIS® Basis-DGM1 370_5812 http://fbinter.stadt-berlin.de/fb/atom/DGM1/370_5812.zip

ATKIS® Basis-DGM1 370_5814 http://fbinter.stadt-berlin.de/fb/atom/DGM1/370_5814.zip

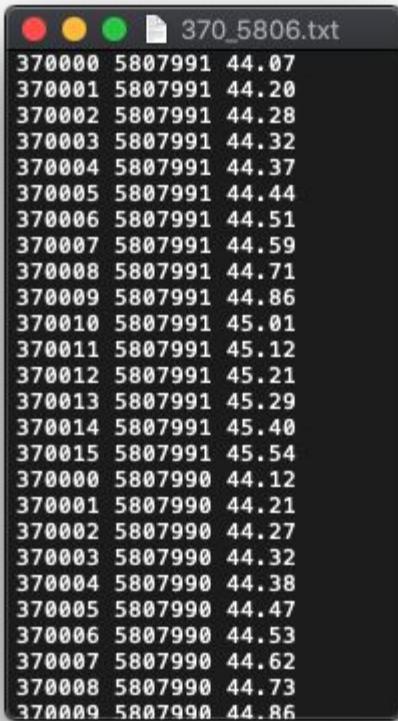
ATKIS® Basis-DGM1 372_5804 http://fbinter.stadt-berlin.de/fb/atom/DGM1/372_5804.zip

Explanation

“Structured”

279 files
~24GB





			370_5806.txt
370000	5807991	44.07	
370001	5807991	44.20	
370002	5807991	44.28	
370003	5807991	44.32	
370004	5807991	44.37	
370005	5807991	44.44	
370006	5807991	44.51	
370007	5807991	44.59	
370008	5807991	44.71	
370009	5807991	44.86	
370010	5807991	45.01	
370011	5807991	45.12	
370012	5807991	45.21	
370013	5807991	45.29	
370014	5807991	45.40	
370015	5807991	45.54	
370000	5807990	44.12	
370001	5807990	44.21	
370002	5807990	44.27	
370003	5807990	44.32	
370004	5807990	44.38	
370005	5807990	44.47	
370006	5807990	44.53	
370007	5807990	44.62	
370008	5807990	44.73	
370009	5807990	44.86	



→ ~125GB GeoJSON 



1	2	3	4	5	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
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

Mean ground level

25x “compression”



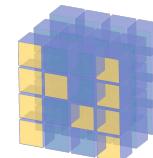
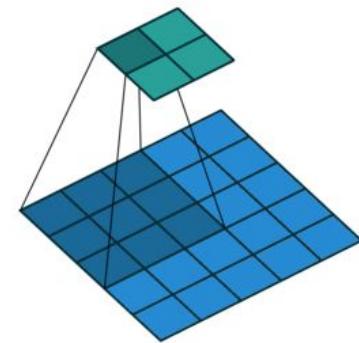
1	2
3	4

⇒ 3.5 h/file



1	2	3	4	5	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
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

Convolution



NumPy

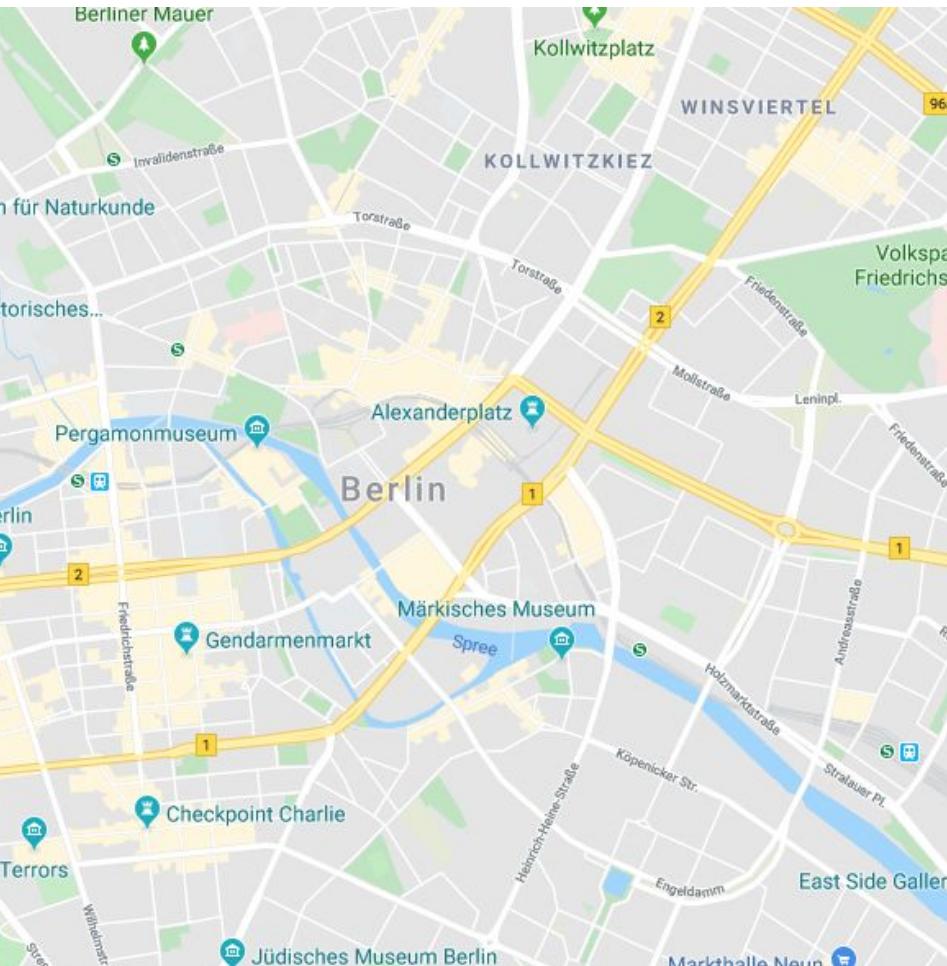
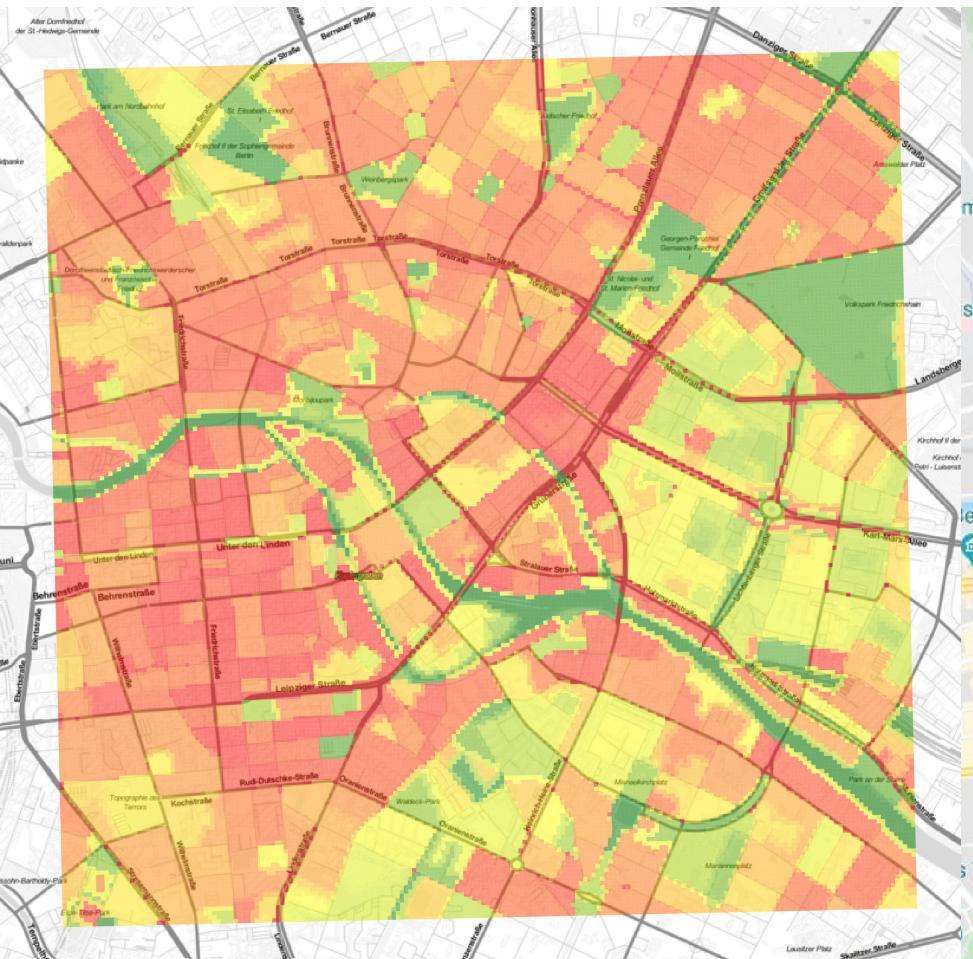
Downloader Package

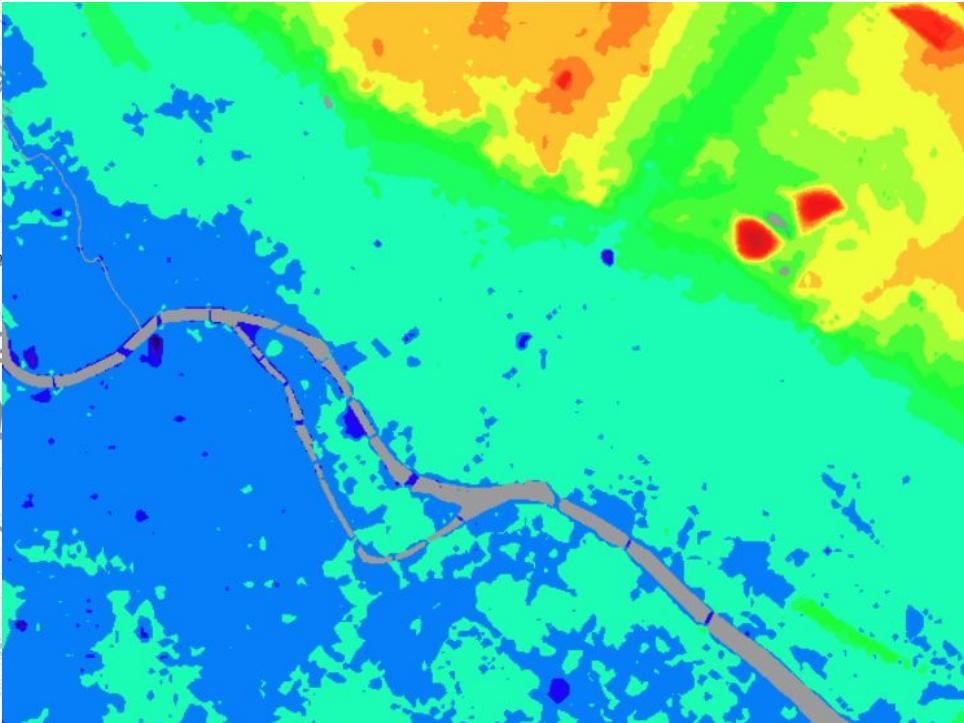
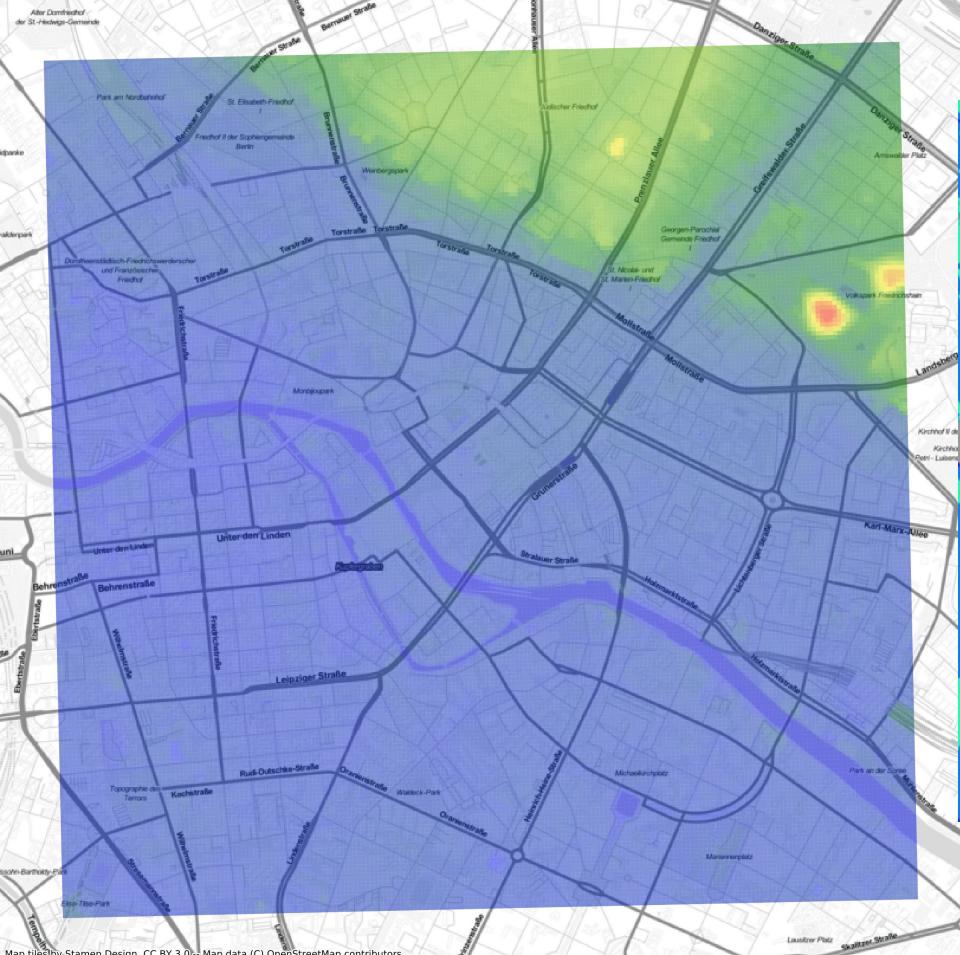
<https://github.com/se-jaeger/berlin-gelaendemodelle-downloader>

```
> pip3 install berlin-opendata-downloader
```

```
> berlin_downloader download ~/berlin_height --compress 5 --file-format geojson
```

⇒ Download + “Compress” in ~2.5h 😊



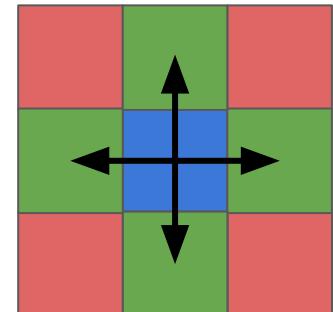


Part 2

The Simulation

Assumptions/Simplifications

- Pixel-based
- Discrete timesteps
- Rainfall is constant in time and geographical dimension
- Water only flows parallel to the axis +
Max. distance of 1 pixel



Goal: Calculate spatial water levels during time.

for n timesteps do:

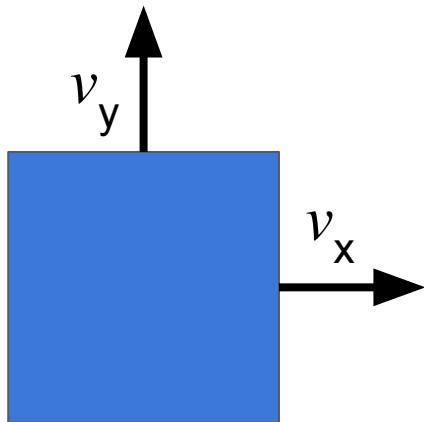
 add rainfall;

 subtract infiltrated water;

 calculate water movement;

 update water level;

How many % water leaves a pixel in which direction?



$$\text{distance}_x = v_x * \text{timestep}$$
$$\text{distance}_y = v_y * \text{timestep}$$

Water Flow According to Gauckler-Manning-Strickler

$$v_m = k_{st} \cdot \sqrt[3]{R^2} \cdot \sqrt{I}$$

Surface	k_{st} as $\frac{m^{\frac{1}{3}}}{s}$
Smooth concrete	100
Straight watercourse	30 - 40
Meandering riverbed with ground vegetation	20 - 30
Torrent with scree	10 - 20
Torrent with undergrowth	< 10

$$R = \frac{\text{water level } (l)}{\text{length} * \text{width} * 1000} = \frac{\text{water level } (l)}{\text{pixel square meter} * 1000}$$

water level

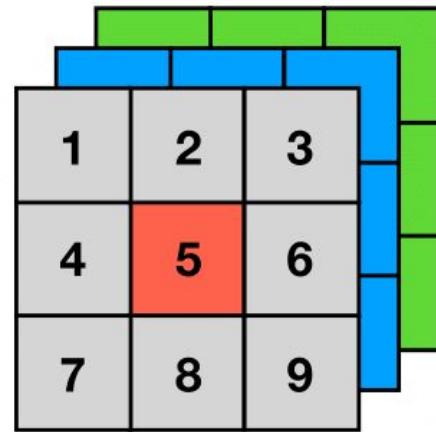
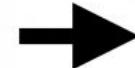
1	2	3
4	5	6
7	8	9

flow x

0	0	0
+1	0.5	-1
0	0	0

flow y

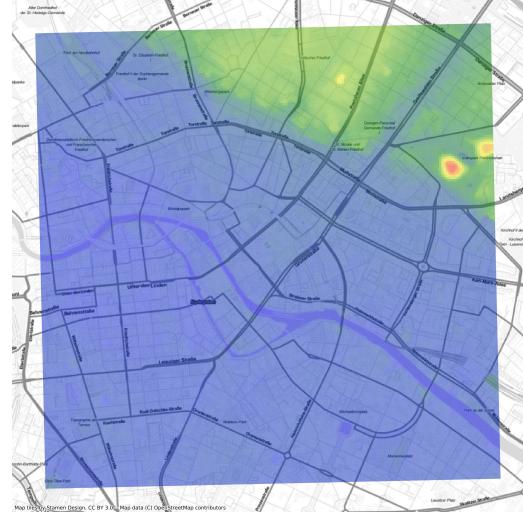
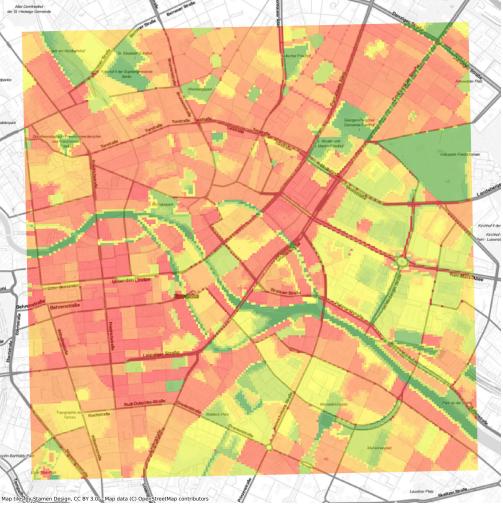
0	-1	0
0	0.5	0
0	+1	0



Berlin '17 - Precipitation >200l/sqm/24h

Simulation: <https://bit.ly/2RkmnB2>

Code: <https://github.com/se-jaeger/urban-technologies-berlin>



Thank You!

Questions?