COMPUTER VISION

MOTION CAPTURE

GOALS:

Taks 1:

- Extract and plot data

Taks2:

- Fill missing data via filtering

Taks 3:

- Import and work on animation data on Unreal Engine 5
- Extract data from Unreal Engine 5
- Project the 3D UE5 animation data on the 2D UE5 animation frame

Optional Task:

- Visualize the BHV file in Blender

Task 1: Extract and plot data

For the extraction of BVH and C3D data, we used the ad hoc developed libraries **BVH-reader** and **Py-C3D**, respectively.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Format Version	1.23	Take Name	ProvaRigidBody	Take Notes		Capture Frame Rate	360	Export Frame Rate	360	Capture Start Time	2024-03-27 03.38.09	Capture Start Fram
2													
3		Type	Rigid Body	Rigid Body	Rigid Body	Rigid Body	Rigid Body	Rigid Body	Rigid Body	Rigid Body	Rigid Body Marker	Rigid Body Marker	Rigid Body Marker
4		Name	Ragnetto	Ragnetto	Ragnetto	Ragnetto	Ragnetto	Ragnetto	Ragnetto	Ragnetto	Ragnetto:Marker1	Ragnetto:Marker1	Ragnetto:Marker1
5		ID	54A2169DEC4611EI	54A2169DEC4611EI	54A2169DEC4611EI	54A2169DEC4611EI	54A2169DEC4611EI	54A2169DEC4611EI	54A2169DEC4611EI	54A2169DEC4611EI	54A2169DEC4611EI	54A2169DEC4611EI	54A2169DEC4611
6			Rotation	Rotation	Rotation	Rotation	Position	Position	Position	Mean Marker Error	Position	Position	Position
7	Frame	Time (Seconds)	X	Y	Z	W	X	Y	Z		X	Y	Z
8	0	0	0.149361	-0.568893	-0.043389	0.807571	-0.087448	0.994073	-0.794972	0.001122	0.042925	0.990646	-0.74145
9	1	0.002778	0.143539	-0.567756	-0.043152	0.809437	-0.087959	0.994178	-0.794836	0.001235	0.042627	0.990541	-0.74185
10	2	0.005556	0.137857	-0.566945	-0.044355	0.810927	-0.088511	0.99475	-0.794814	0.0014	0.042246	0.990495	-0.74230
11	3	0.008333	0.134579	-0.565902	-0.044334	0.812206	-0.088859	0.994834	-0.794698	0.00144	0.042063	0.990442	-0.7426
12	4	0.011111	0.131117	-0.565062	-0.044123	0.813368	-0.089295	0.994937	-0.794686	0.001601	0.041765	0.990436	-0.74295
13	5	0.013889	0.127337	-0.563432	-0.044027	0.815102	-0.089715	0.995028	-0.794467	0.001524	0.041582	0.990402	-0.74335
14	6	0.016667	0.121409	-0.560986	-0.042141	0.817789	-0.090271	0.994774	-0.793994	0.001337	0.041371	0.990408	-0.74375
15	7	0.019444	0.115289	-0.559567	-0.040756	0.819715	-0.090674	0.994807	-0.793828	0.001531	0.041184	0.990496	-0.74414
16	8	0.022222	0.112227	-0.558671	-0.04065	0.820755	-0.091059	0.994889	-0.793733	0.001642	0.040926	0.990451	-0.74440
17	9	0.025	0.107876	-0.558605	-0.04069	0.821381	-0.091852	0.994942	-0.793691	0.001881	0.040169	0.990219	-0.74448
18	10	0.027778	0.10396	-0.556442	-0.03901	0.823434	-0.092169	0.99495	-0.793374	0.002057	0.040131	0.990515	-0.74489
19	11	0.030556											
20	12	0.033333	0.091984	-0.551829	-0.038252	0.827986	-0.093218	0.996018	-0.792773	0.001249	0.039672	0.991164	-0.74597
21	13	0.036111											
22	14	0.038889											

CSV RIGID BODY STRUCTURE

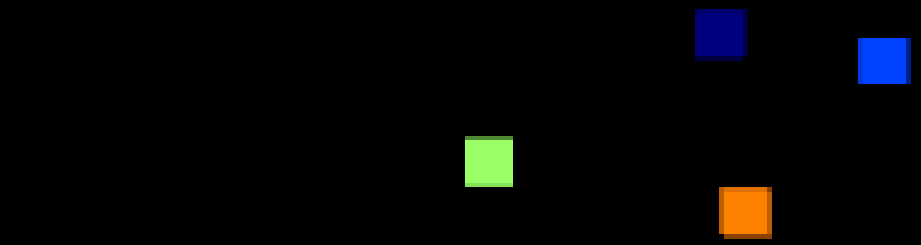
It consists of coordinate columns divided in time stamps, for both marker quaternions and marker location.

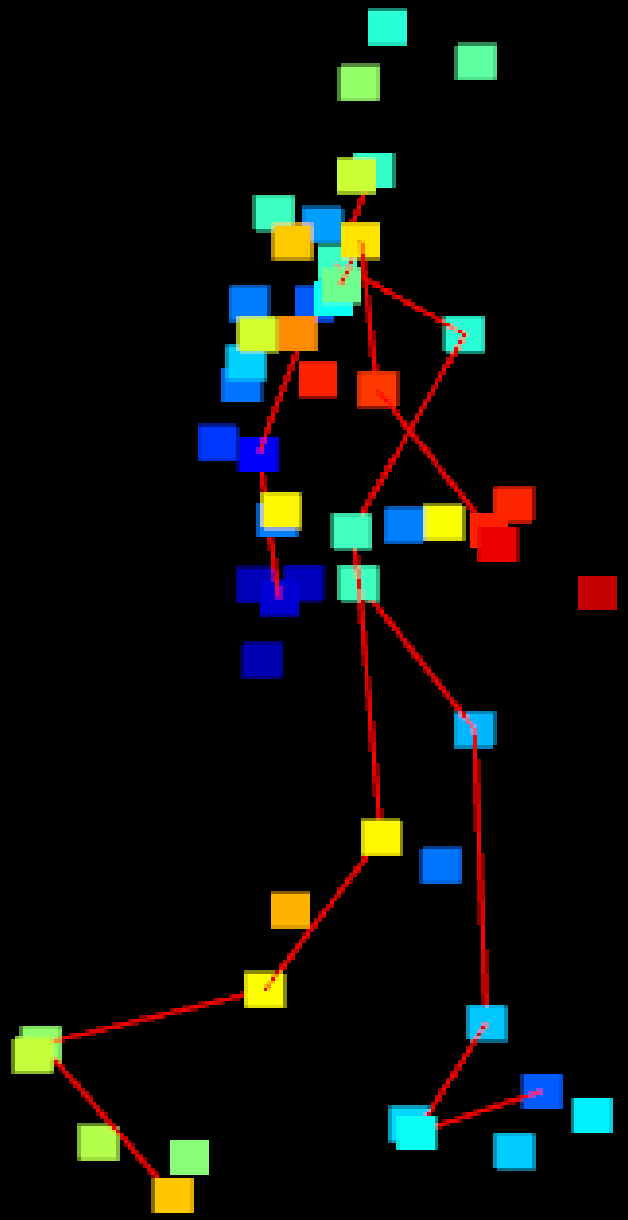
	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Format Version	1.23	Take Name	LucaTest	Take Notes		Capture Frame Rate	360	Export Frame Rate	360	Capture Start Time	2024-03-27 04.12.19	Capture Start F
2													
3		Type	Bone	Bone	Bone	Bone	Bone	Bone	Bone	Bone Marker	Bone Marker	Bone Marker	Bone Marker
4		Name	Luca:Hip	Luca:Hip	Luca:Hip	Luca:Hip	Luca:Hip	Luca:Hip	Luca:Hip	Luca:WaistLFront	Luca:WaistLFront	Luca:WaistLFront	Luca:WaistRFro
5		ID	1	1	1	1	1	1	1	5	5	5	
6			Rotation	Rotation	Rotation	Rotation	Position	Position	Position	Position	Position	Position	Position
7	Frame	Time (Seconds)	X	Y	Z	W	X	Y	Z	X	Y	Z	X
8	0	0	0.01975	-0.676201	-0.023314	-0.736083	0.427239	0.853169	-0.074397	0.519527	0.938742	-0.201843	0.50
9	1	0.002778	0.019631	-0.675991	-0.023532	-0.736272	0.428557	0.853416	-0.074021	0.520902	0.939066	-0.201375	0.50
10	2	0.005556	0.019235	-0.675808	-0.023642	-0.736448	0.429934	0.85367	-0.073675	0.522373	0.939381	-0.20092	0.50
11	3	0.008333	0.018923	-0.675745	-0.024128	-0.736498	0.431296	0.853929	-0.073512	0.523726	0.93981	-0.200649	0.51
12	4	0.011111	0.018581	-0.675766	-0.02447	-0.736476	0.432622	0.854119	-0.073305	0.525037	0.940127	-0.200367	0.51
13	5	0.013889	0.018331	-0.675628	-0.024928	-0.736593	0.434096	0.854307	-0.073082	0.526523	0.940473	-0.200028	0.51
14	6	0.016667	0.017771	-0.675278	-0.025323	-0.736914	0.435456	0.854482	-0.072807	0.528014	0.940807	-0.199549	0.51
15	7	0.019444	0.017815	-0.675389	-0.025603	-0.736801	0.436814	0.854733	-0.072568	0.52929	0.94114	-0.199314	0.51
16	8	0.022222	0.01787	-0.675304	-0.026361	-0.736852	0.438366	0.854908	-0.072231	0.53076	0.941545	-0.198879	0.51
17	9	0.025	0.017403	-0.675212	-0.026475	-0.736943	0.439726	0.855122	-0.072021	0.53219	0.941824	-0.198574	0.51
18	10	0.027778	0.017262	-0.675216	-0.02678	-0.736932	0.441179	0.855296	-0.07175	0.533615	0.942101	-0.198253	0.52
19	11	0.030556	0.017182	-0.6753	-0.026904	-0.736852	0.442576	0.855408	-0.071476	0.534975	0.942255	-0.197977	0.52
20	12	0.033333	0.01668	-0.675208	-0.027163	-0.736938	0.443884	0.855557	-0.071258	0.536337	0.942515	-0.197644	0.52
21	13	0.036111	0.016536	-0.67509	-0.027585	-0.737034	0.445371	0.855729	-0.070968	0.537822	0.942826	-0.197259	0.52
22	14	0.038889	0.016533	-0.674977	-0.027951	-0.737124	0.446732	0.855859	-0.070662	0.539171	0.94307	-0.196883	0.52

CSV SKELETON STRUCTURE

Like the rigid body file, it consists of coordinate columns divided in time stamps.
It has two different set of markers.

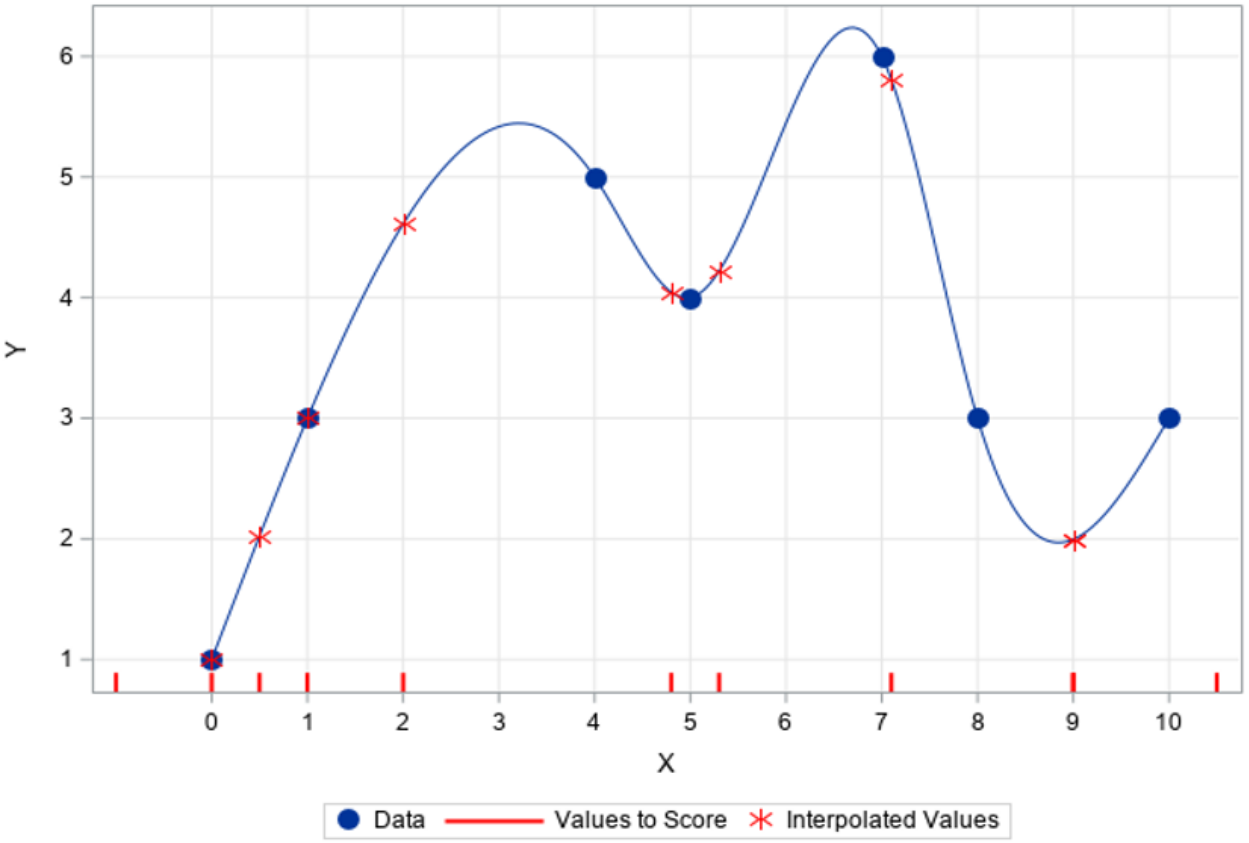
Rigid body and skeleton plot result



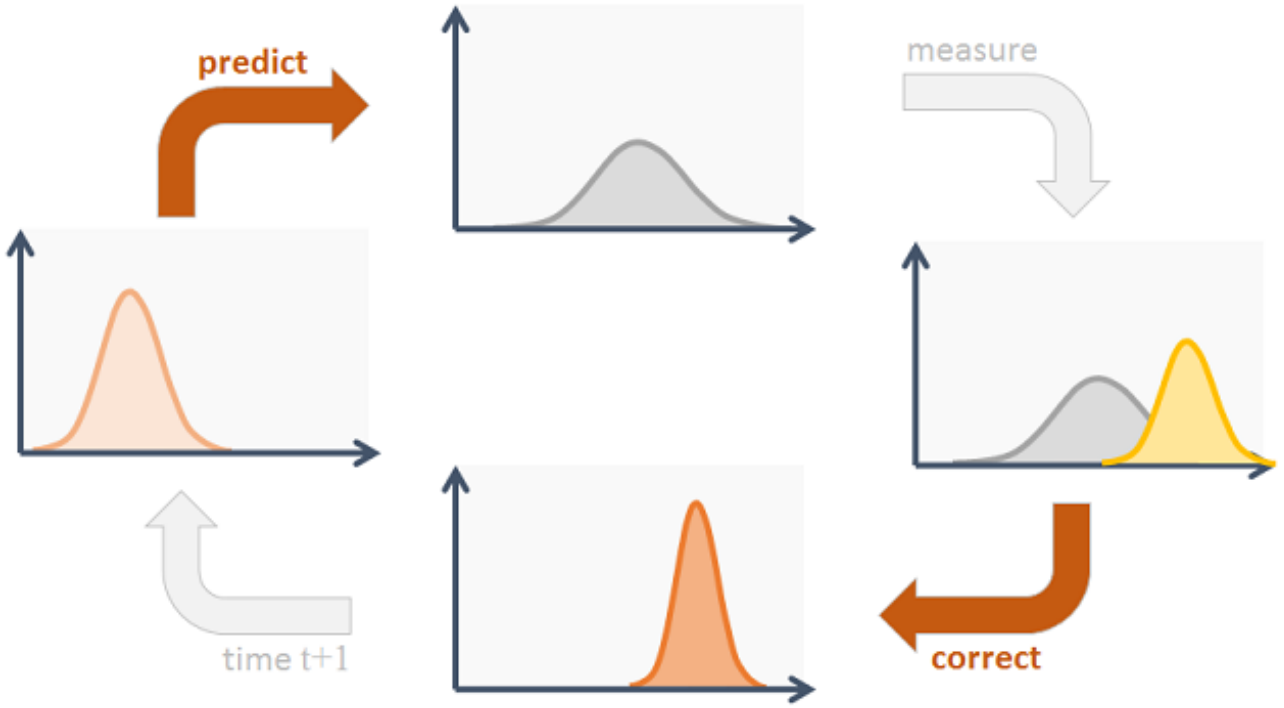


Task 2: Fill missing data via filtering

Cubic spline interpolation

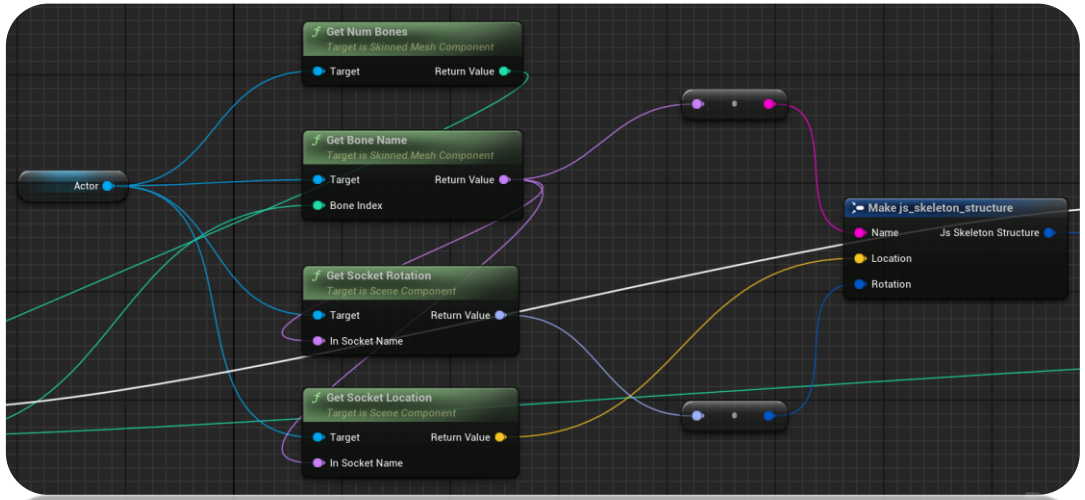


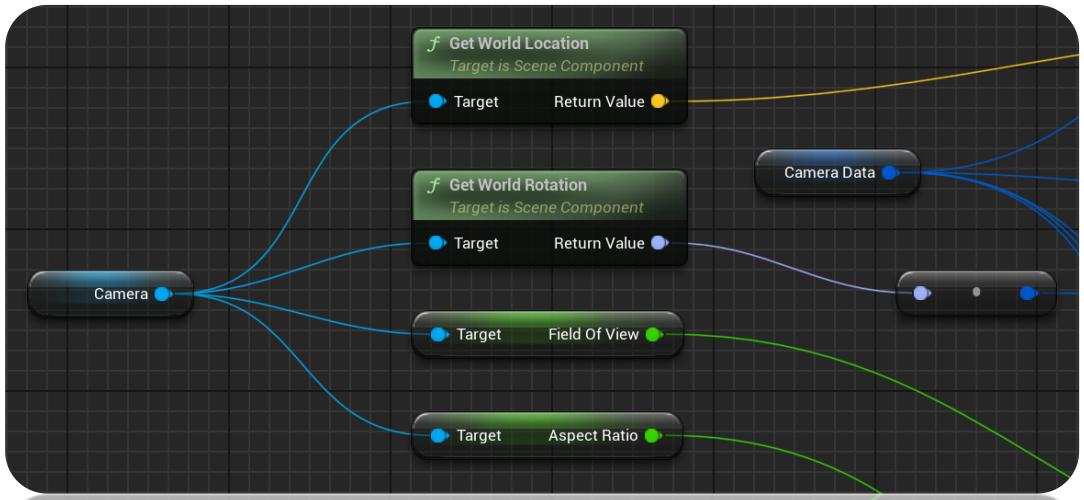
Kalman filter



Task 3: Motion position estimation

Scene setup





Data collection

JSON files structures for skeleton points and camera data

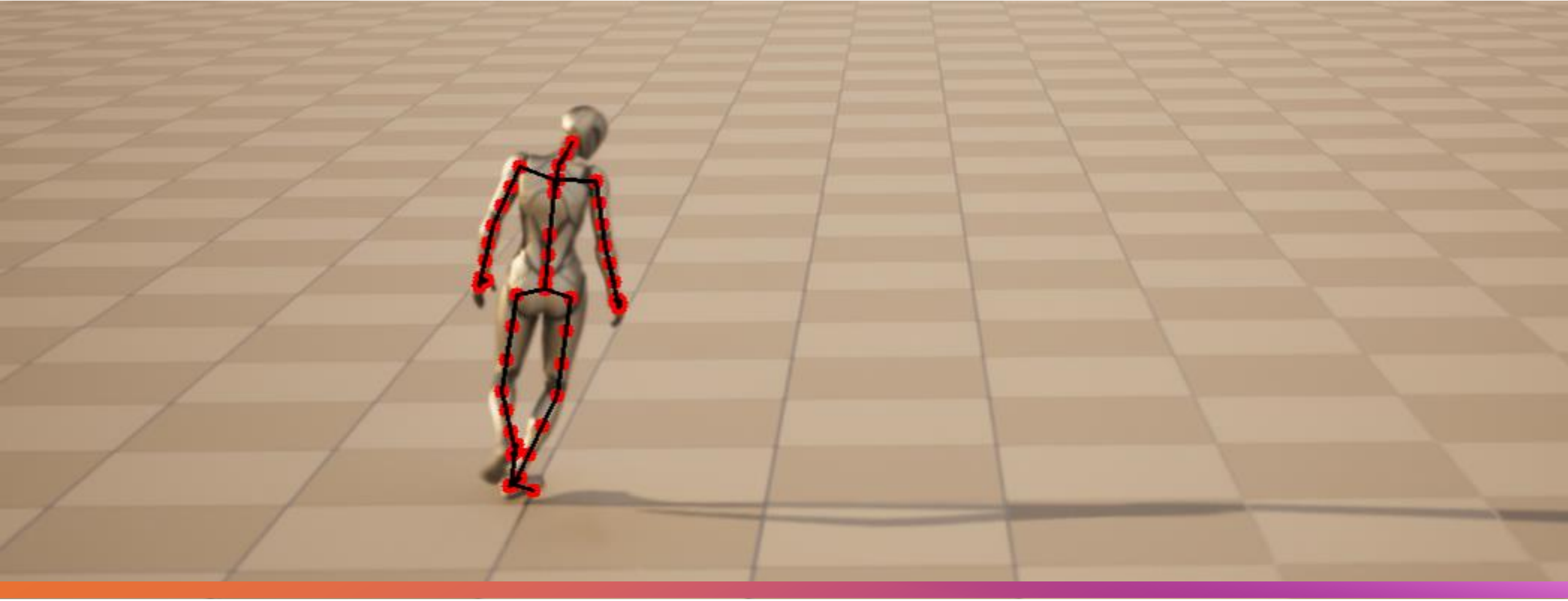

```
        {
            "name": "pelvis",
            "location": {
                "x": 49.559196472167969,
                "y": -8.5577392578125,
                "z": 98.693283081054688
            },
            "rotation": {
                "x": -0.44846883445195546,
                "y": -0.55082654988645996,
                "z": -0.50616323980659506,
                "w": 0.48914679916562454
            }
        },
        {
            "name": "spine_01",
            "location": {
                "x": 49.264496651105581,
                "y": -8.5605056953667873,
                "z": 101.14750428215508
            },
            "rotation": {
                "x": -0.50803821430808871,
                "y": -0.52993741658048599,
                "z": -0.44822552520030051,
                "w": 0.51005625755358042
            }
```

```
{"world_location":{"x":1145.6029254272141,"y":42.792060594793874,"z":308.13871621669489},
"world_rotation":{"x":0.08186823989787205,"y":0.0035490366923341325,"z":-0.99663594541313849,"w":0.0013371406210118569},
"field_of_view":37.497356414794922,"aspect_ratio":1.7777777910232544}
```


You, 1 second ago • Uncommitted changes

Project 3D points on 2D frame

- We switched from UE5 left-handed coordinate system to OpenCV right-handed coordinate system
- We used OpenCV functions to project and plot the points

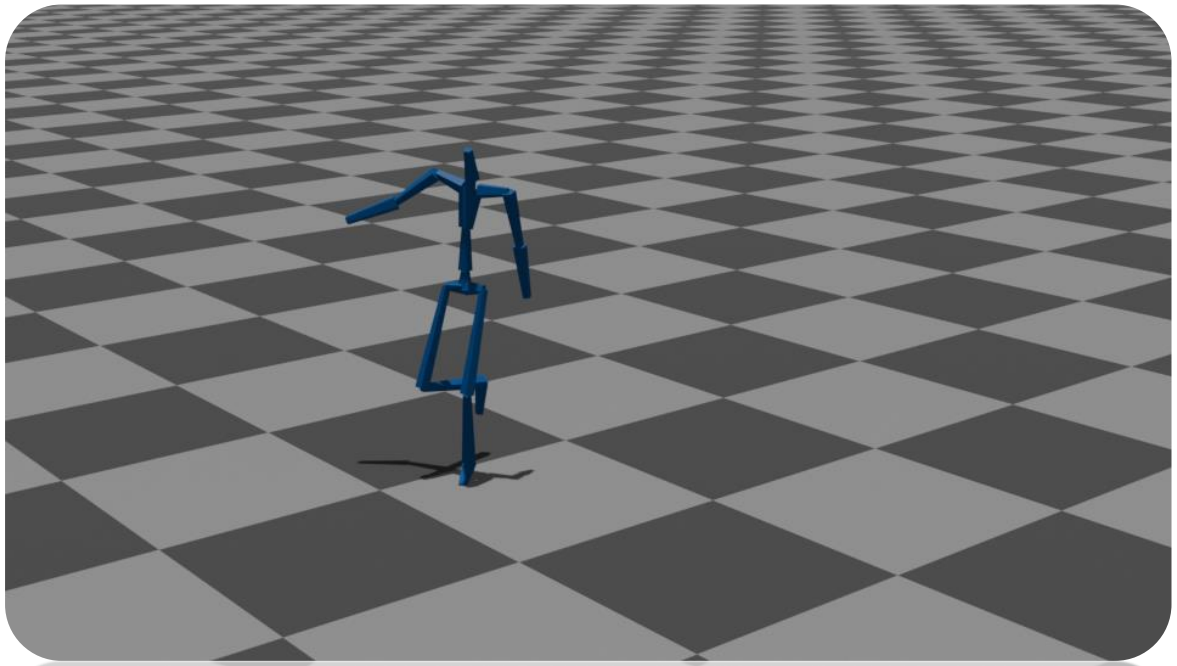


Skeleton projection result

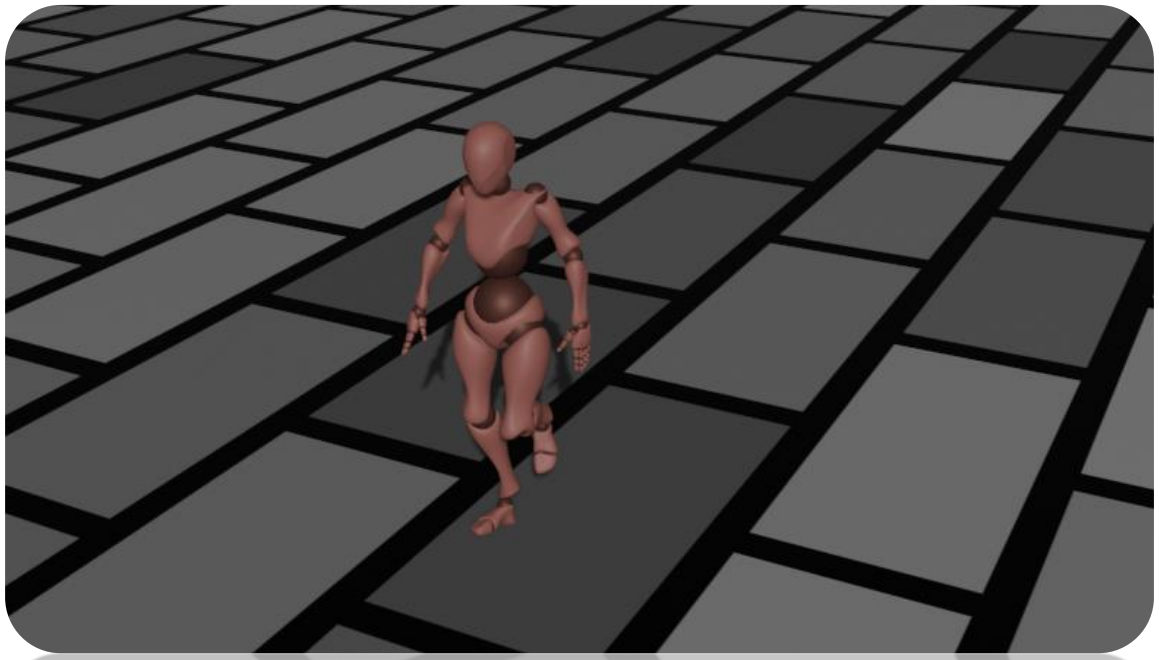
Optional Task: Render BVH file in Blender

To render the provided BVH file in Blender we used **Python APIs** and the **Deep-Motion-Editing** library.

Rendering

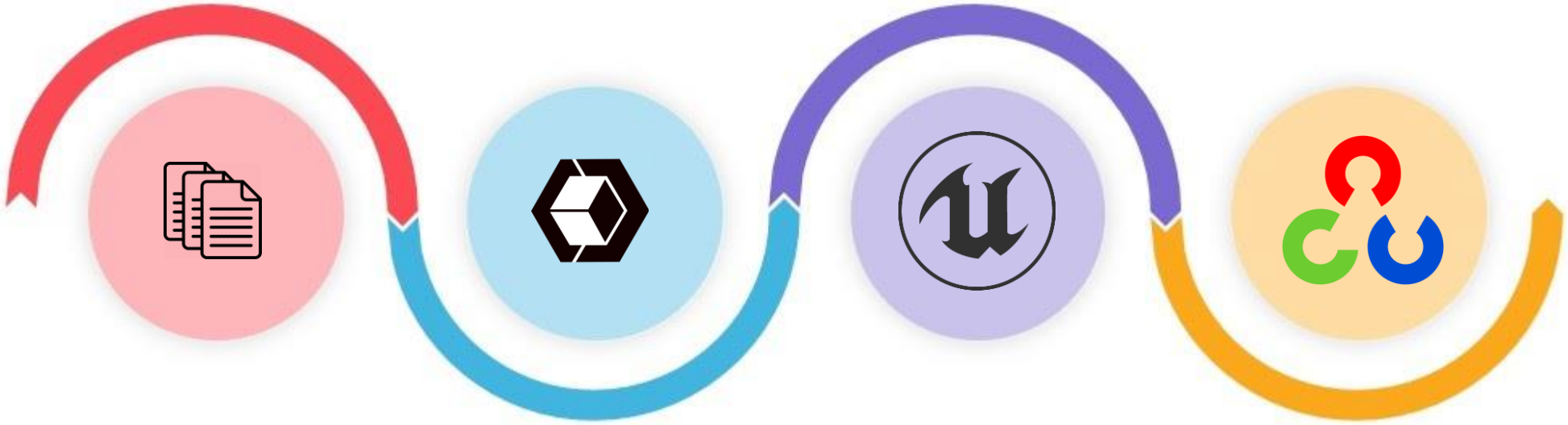


Skinning



Blender rendering

Conclusions



Thank you