



# Computer Vision

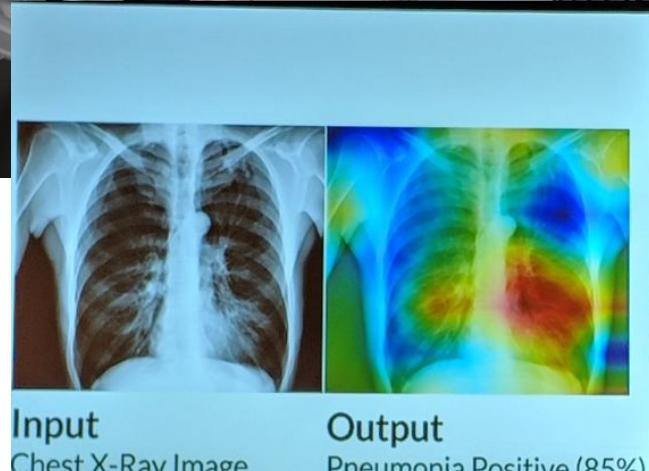
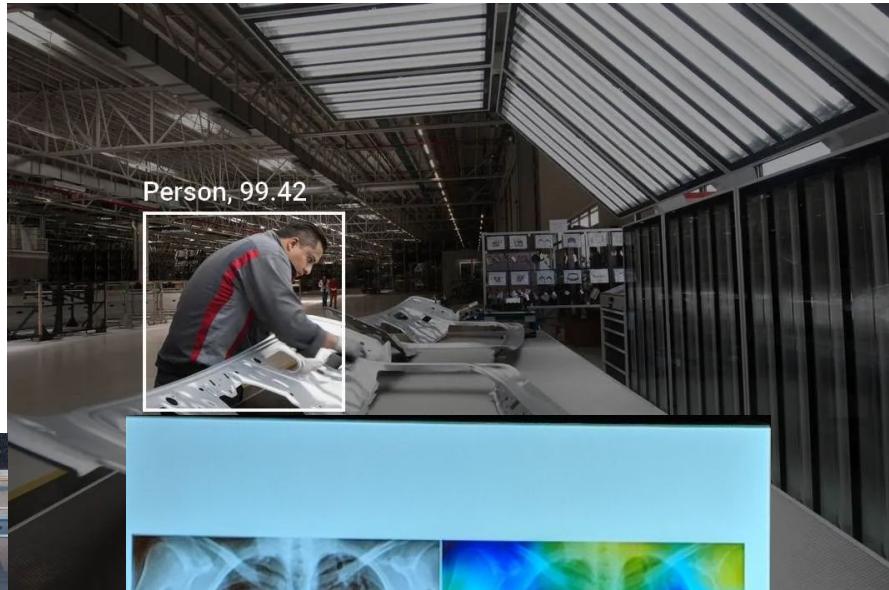
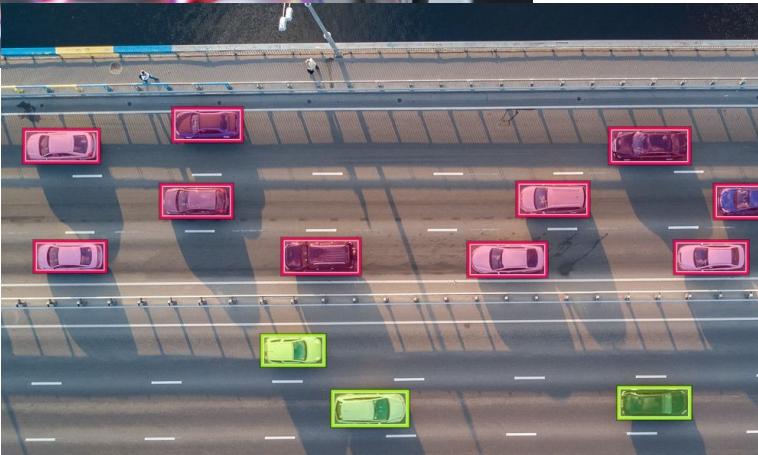
Via OpenCV

# Dette er et computer vision kursus!

Sproget er valgfrit og “bare” et værktøj



# Aktuelle computer vision use cases



# Undervisers baggrund

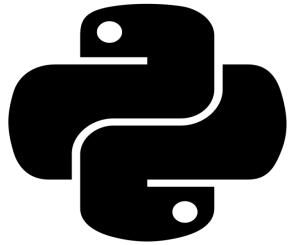
- [AMNI1@ucl.dk](mailto:AMNI1@ucl.dk)
- Cand.it IT & kognition
- AI konsulent for Front AI
- Ekstern konsulent hos Segtnan
- Aktuelt Adjunkt på UCL (4 år)
- Aktuelt freelancer i computer vision (3 år)
  - Finde små tumorer i forbindelse med pankreaskræft

**front.ai**

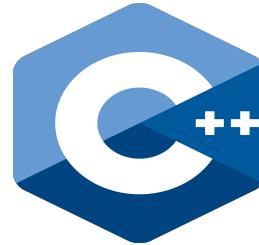


UNIVERSITY OF  
COPENHAGEN

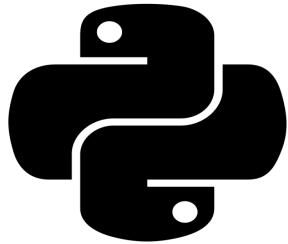




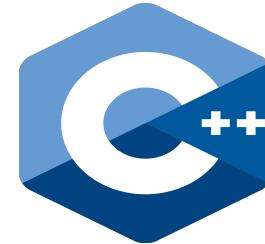
- Interpreted
- Svage typer (loosely typed)
- Langsommere
- Højniveau (ingen direkte adgang til hardware)



- Kompileret (compiled)
- Stærke typer (strongly typed)
- Hurtig
- Lavniveau (direkte adgang til hardware)



- Bruges næsten udelukkende i machine learning
- Bruges til at fortælle et framework hvordan det kan f.eks. træne en model, definere en pipeline, processere data osv.
- Se det som et mere abstrakt sprog til styring af processer, men ikke kontrol af algoritmer og hardware



- Anvendes i realtidssystemer
- Bruges til at skrive optimeret kode med meget kontrol over algoritmen og hardware
- F.eks. er de fleste processeringsalgoritmer skrevet i C++, men kan kaldes via Python og deslige.

# Make or break - Python vs C++

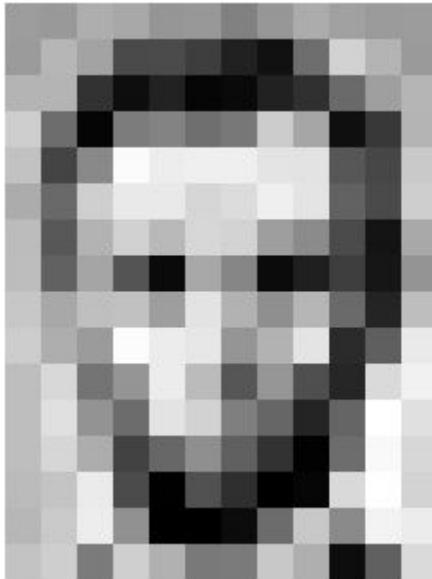
- Punktskyer til 3D modellering egner  
Python sig slet ikke til, da det kræver så  
mange punkter i 3D
- Selve modellerne i machine learning-regi  
er alle skrevet i C grundet performance
- Shaders, raytracing, lys generelt, er alt  
sammen C
- Python er super nyttigt til at beskrive  
pipelines og algoritmer; tager længere tid i  
C/C++

# Hvorfor bør du lære Computer Vision?

- CV er vidt anvendt i industrien, bl.a. i robotics, embedded og spilindustrien
- For bedre at forstå computergrafik generelt
- Det er et af de få felter i software der både er gammelt og stadig i rivende udvikling

Tid til lidt Computer vision  
basics!

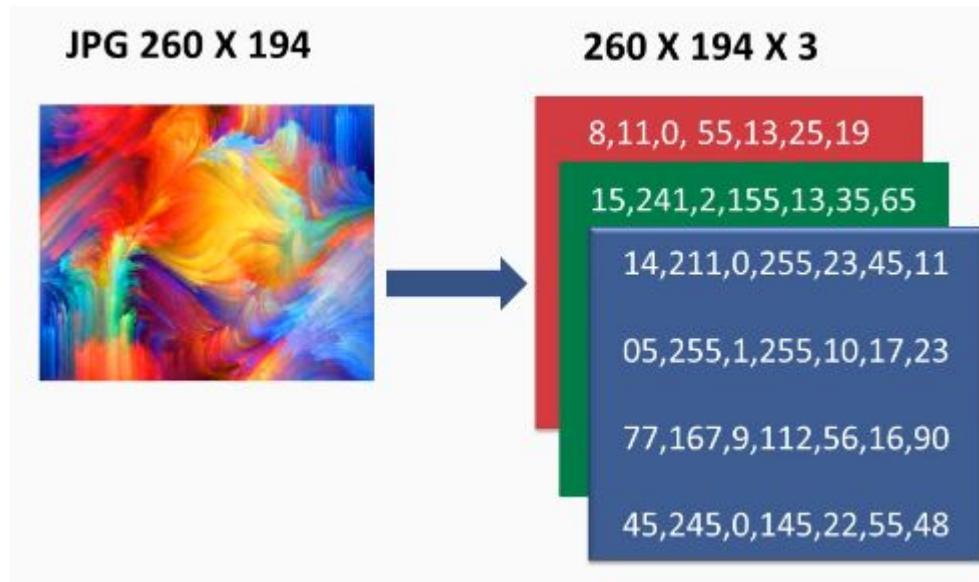
# Hvad er pixels og billeder? Gråtoner først.



157	163	174	168	150	162	129	151	172	161	155	166
165	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	84	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	297	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	105	36	190
206	174	155	252	236	231	149	178	228	49	95	234
190	216	116	149	236	187	85	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	95	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
196	206	123	207	177	121	123	209	175	13	96	218

157	153	174	168	150	152	129	151	172	161	156	156
165	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	84	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	105	36	190
206	174	155	252	236	231	149	178	228	49	95	234
190	216	116	149	236	187	85	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	95	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
196	206	123	207	177	121	123	209	175	13	96	218

# RGB billeder



# Hvad er et “godt billede” til computer vision?

- Høj kontrast - god linse og hardware er ikke nok
- Lav forvrængning - en tripod kan være vigtig
- Nok lys - det meste vi ser er reflektioner
- Høj oplosning - til et vist punkt
- Ikke meget støj - vi snakker støj ud fra et signalbehandlingsperspektiv

# Lille øvelse

- Gå sammen 3-4 stykker
- Tag minimum 3 billeder i forskellige lysforhold og af forskellige objekter
- Vurdér dem hvor i gennemgår kriterierne fra sidste slide

# Den typiske proces for detektion

- Start med et godt billede
- Filtrér overflødige ting væk - high-pass, low-pass, thresholding
- Highlight de vigtige objekter - kontrastforhold, Gabor filter, morfologi, farveboost
- Manipulér billedet så det er nemmere for detektionsalgoritmen - blurring, morfologi
- Afprøv detektion

# Porteføljeeksamen

- Se det som en logbog/journal, eller en mest visuel step-by-step over dine eksperimenter (hvordan du behandlede billedet, hvilke metoder har du anvendt på dem)
- Formatet er sådan så i kan begynde allerede i dag uden at skulle omskrive senere.

# Porteføljens format og indhold

- Indledning med valgt case og problemstilling
- Et diagram f.eks. Flowchart over din endelige løsning/pipeline
- Før læsere step-by-step gennem hver node i dit flowchart, med før og efter billeder for hver metode
- Evt. diskussion om hvad du tænker kan forbedres og hvordan
- Aflever som PDF
- Resten beskrevet i mappen porteføljelekav

Tag billeder

Tid til OpenCV og prøve lidt computer  
vision!

# Basal billedanalyse

Dagens opgaver i overskrifter:

- Gennemsnittet af et gråtonebillede
- Gennemsnittet af hver kanal af et RGB billede
- Histogrammer på billeder
- Sammenligning af billeder baseret på farve og intensitet

Disse opgaver er uddybet i ugens opgavesæt.