

FASHION AND APPAREL CLASSIFICATION USING CONVOLUTIONAL NEURAL NETWORKS

Alexander Schindler

alexander.schindler@ait.ac.at

Thomas Lidy

lidy@ifs.tuwien.ac.at

Stephan Karner

stephan.karner@monstyle.io

Matthias Hecker

matthias.hecker@monstyle.io

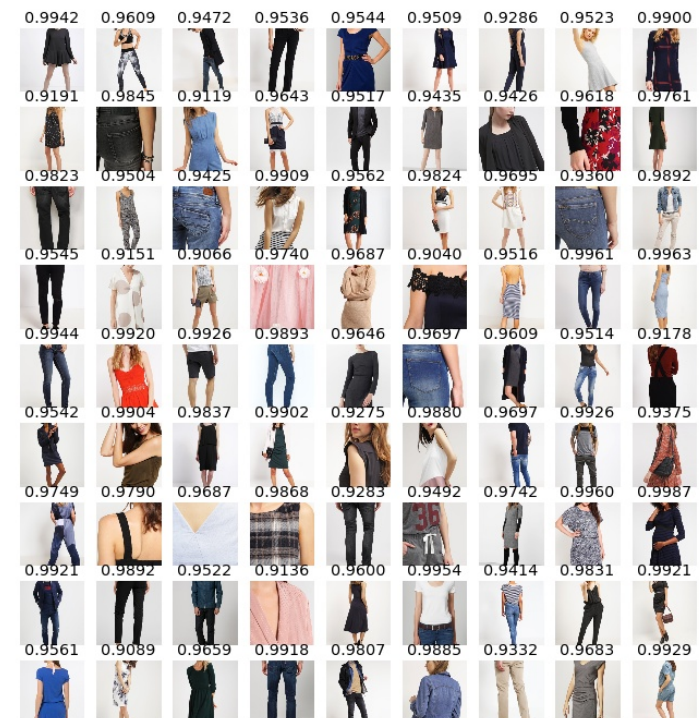


FASHION IMAGE CLASSIFICATION

- Online e-commerce access to product images
 - Asos-EU, Farfetch, Zalando
 - Images & metadat
- **Problem**
 - Metadata differs in
 - Quality, granularity, taxonomy
 - Taxonomy varies in depth of categorical hirarchy
- **Task**
 - use CNNs to
 - Consolidate Metadta
 - Enrich Metadata

BRIEF OVERVIEW

- Empirical study
- Applying deep Convolutional Neural Networks to fashion classification
- Evaluated five CNN architectures
- Custom and pre-trained models
- Evaluated on three tasks
 - Person detection
 - Product classification
 - Gender prediction



DATASETS

• Person

- 7.833 images
 - 5.669 labeled as persons
 - 2.164 labeled as products

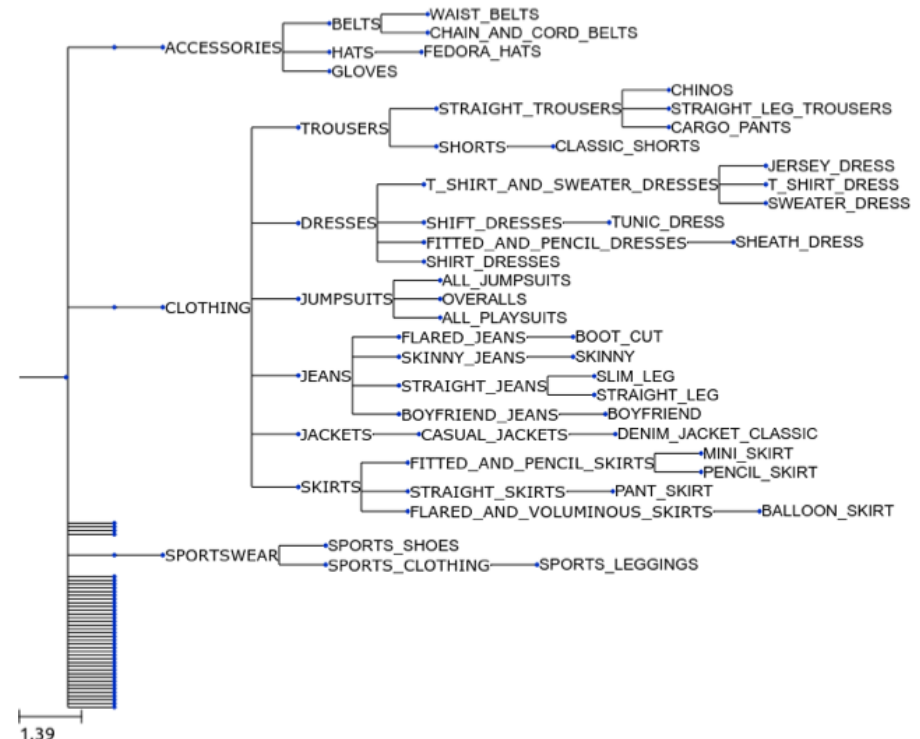
• Products

- 234.884 images
- 39.474 products
 - ~5,95 images per product
- Ground-truth labels assignments
 - Product category
 - Label hierarchy
 - Gender
 - Age

Person



Product



DATA QUALITY / ISSUES

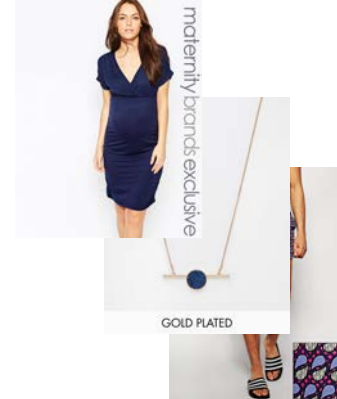
White background



Worn by persons



Text, Overlays



Close-up texture



Close-up fit



Multiple objects



Brand logo

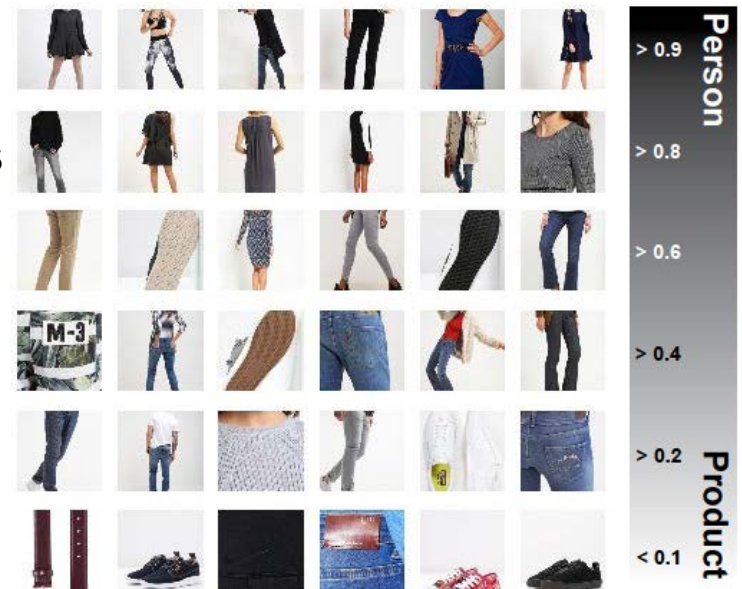


Misc



PERSON DETECTION

- **Products also presented by persons**
 - How they look when worn?
- **Problem**
 - Person wears multiple products
 - Single-label classification
 - Decision problem
- **Approach**
 - Train model to identify persons
 - Use model to filter images with persons
 - VGG-like custom model
- **Results**
 - **91.07%** accuracy on persons dataset



PRODUCT CLASSIFICATION

- **Deep Neural Network Architectures**

- Vgg16 and Vgg19
- InceptionV3
- Custom CNN and Vgg-like

- **Experiments**

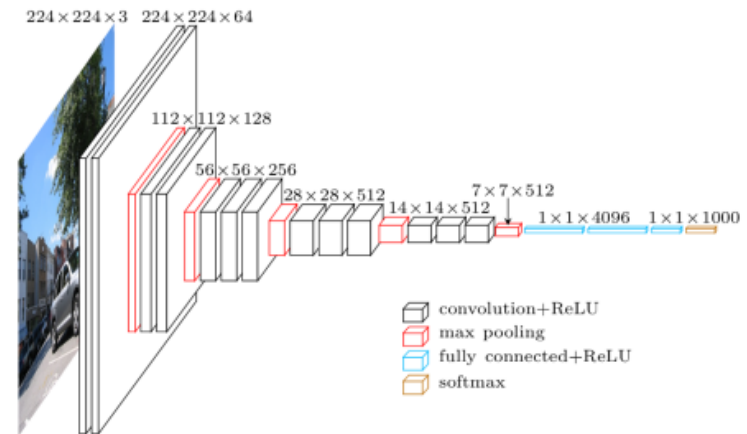
- From-Scratch
- Pre-Trained

- **Evaluation**

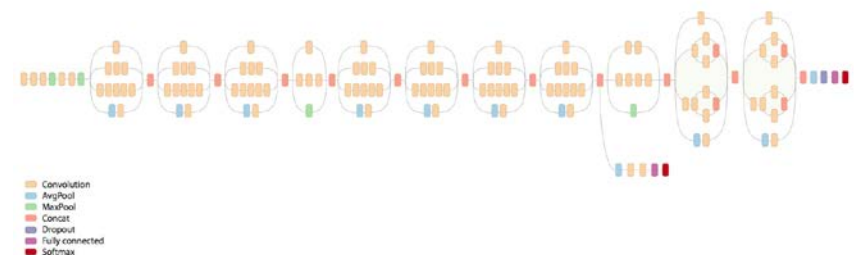
- 4-Fold Crossvalidation
- Grouped Stratification

- **Metrics**

- Raw Accuracy
- Max of Sum per product



David Fossard, <https://www.cs.toronto.edu/~frossard/post/vgg16/>



John Shlens, <https://research.googleblog.com/2016/03/train-your-own-image-classifier-with.html>

EXPERIMENTAL SET-UP

- **Small scale**
 - Subset of 23.305 images
- **Large scale**
 - 234.408 images
- **All Models**
 - Data Augmentation
 - 25% vertically and horizontally shifting
 - 25% zoom range
 - Horizontal flipping

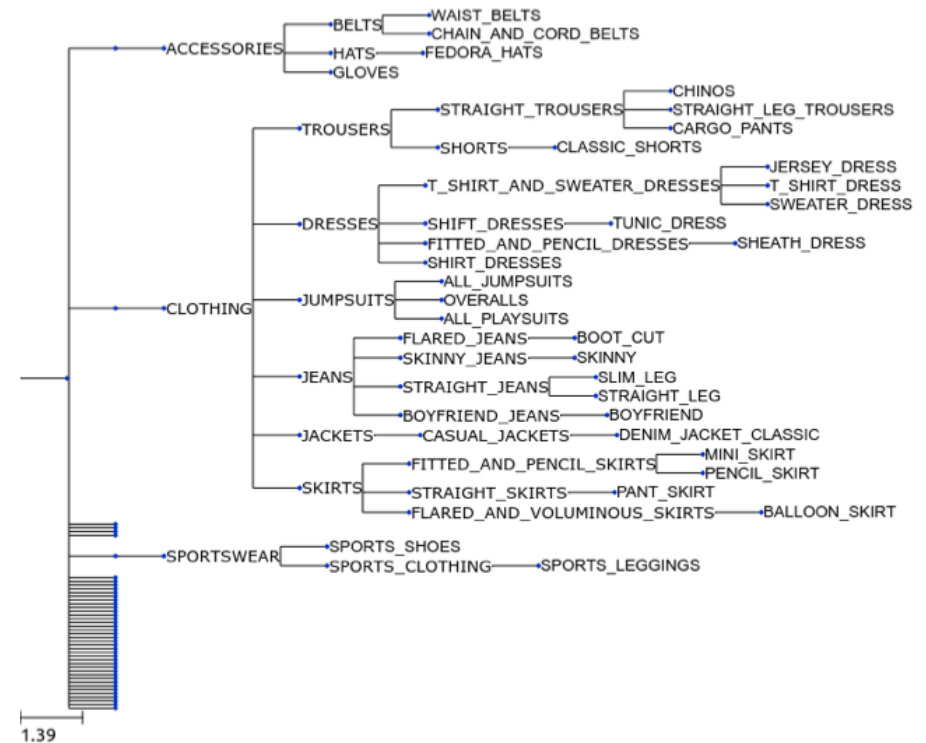
RESULTS – SMALL SCALE (24K)

- **Best results:** Pre-trained + fine-tune entire model
 - Freezing network + training only top layers not as good
- **Person filter** did not improve performance
- Small custom models have advantage of speed, but not as accurate

Description	best fold	best fold cum max	Mean cum max
InceptionV3, pretrained, fine-tuned	0.706	0.794	0.791
InceptionV3, pretrained, fine-tuned	0.658	0.729	0.716
VGG16, pretrained, fine-tuned	0.646	0.711	0.691
InceptionV3, pretrained, fine-tuned, person filter model as layer	0.569	0.685	0.658
VGG19, pretrained, fine-tuned	0.579	0.673	0.634
InceptionV3, pretrained, fine-tuned, no augmentation	0.564	0.673	0.647
VGG19, pretrained, train only top-layers	0.578	0.669	0.343
VGG16, pretrained, train only top-layers	0.603	0.652	0.368
InceptionV3, pretrained, train only top-layers	0.585	0.650	0.643
InceptionV3, pretrained, fine-tuned - person filtered metadata	0.640	0.636	0.614
InceptionV3, clean	0.492	0.594	0.580
Custom CNN, augmentation	0.506	0.568	0.538
Custom CNN	0.463	0.556	0.523
Custom VGG-like	0.438	0.549	0.519
VGG16, clean	0.439	0.455	0.443
VGG19, clean	0.437	0.447	0.430
VGG19, pretrained, train only top-layers	0.819	0.887	0.880
InceptionV3, pretrained, fine-tuned	0.798	0.863	0.836
VGG19, pretrained, fine-tuned	0.762	0.846	0.830

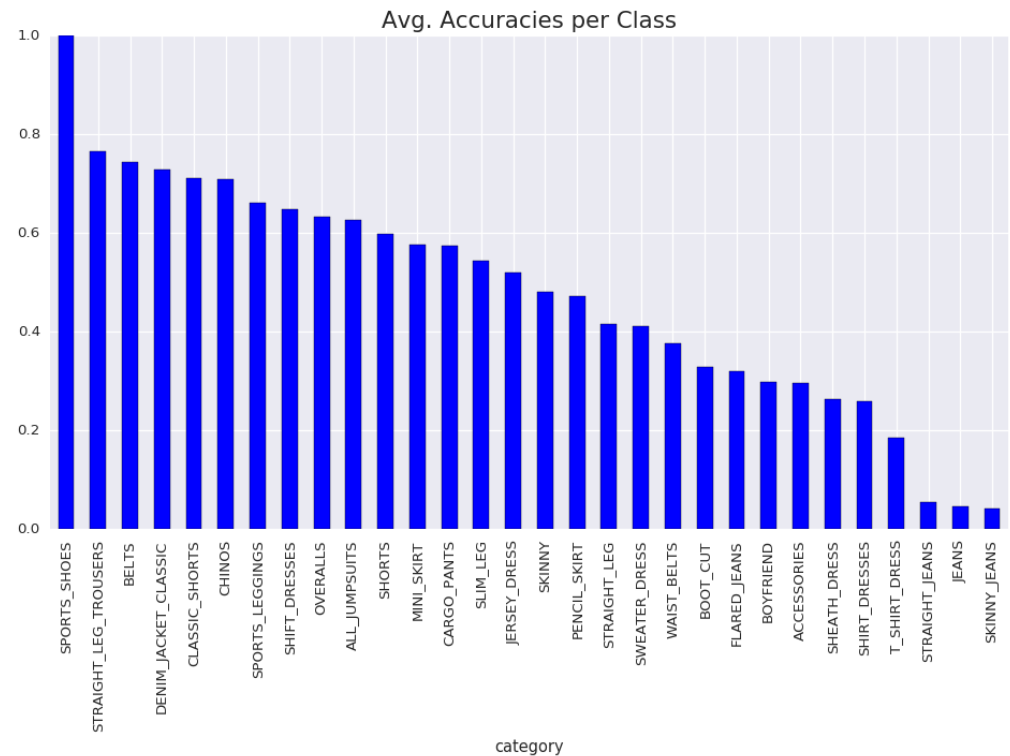
CONFUSIONS – SMALL SCALE (24K)

		Confusion matrix																																
True label	CHINOS	35	2	23	0	0	0	0	4	0	0	0	4	3	0	4	1	11	0	0	1	0	0	1	12	0	0	0	3	1	13			
	CLASSIC_SHORTS	1	47	39	11	4	14	11	42	173	67	14	1	611	51	75	19	1211	289	89	20	33	0	63	15	24	0	14	6	57	6	0		
	ALL_JUMPSUITS	7	7	233	5	0	0	0	1	2	1	0	0	14	2	0	0	2	121	2	1	0	0	2	1	608	0	1	0	1	0	36		
	JERSEY_DRESS	0	6	3	566	62	1	9	4	9	236	4	1	4	7	0	0	2	0	18	99	3	278	0	0	10	398	2	0	0	0	0		
	SPORTS_SHOES	0	21	2	44	951	16	135	20	19	25	13	1	10	42	2	0	5	1	92	123	6	956	2	6	27	670	3	1	1	0	0		
	SHIFT_DRESSES	0	7	1	0	7	479	89	146	3	0	1	3	0	7	0	0	13	1	8	7	1	46	0	1	0	11	2	0	1	0	0		
	BELTS	1	27	1	4	97	48	31	42	254	14	7	67	8	6	9	3	0	13	4	18	49	0	404	4	8	2	111	133	1	0	1	0	
	OVERALLS	2	204	6	1	14	139	288	46	12	9	1	2	141	182	45	36	6	210	18	365	24	0	50	23	34	0	22	30	17	3	0	0	
	SKINNY	0	40	1	5	6	8	17	18	933	7	0	41	12	20	4	0	45	5	9	19	1	50	0	1	0	28	3	5	2	0	0	0	
	SLIM_LEG	0	8	1	298	29	0	1	4	9	311	2	2	2	6	0	0	4	0	16	73	8	103	1	0	3	93	0	1	0	0	0	0	
	STRAIGHT_LEG	0	1	2	1	24	1	68	8	1	3	18	2	0	1	1	0	0	1	1	43	5	143	3	0	8	51	2	1	0	0	0	0	
	JEANS	4	496	9	1	6	0	5	84	36	1	0	6880	106	23	79	626	387	2110	9	14	0	24	23	47	1	2	5	764	8	2	0	0	
	SHIRT_DRESSES	5	50	3	1	9	0	11	164	9	1	0	2211804	12	235	27	382	15	20	10	0	26	16	24	0	4	3	77	0	0	0	0	0	
	CARGO_PANTS	0	88	0	2	25	4	6	34	20	4	0	34	7	898	1	0	59	5	19	23	0	48	2	5	0	17	1	1	1	1	1	0	
	SHEATH_DRESS	1	34	3	0	5	1	11	47	7	2	0	214368	2	982	50	209	6	7	15	0	10	6	13	0	2	7	76	1	1	1	1	0	
	STRAIGHT_LEG_TROUSERS	0	108	1	0	1	0	1	13	2	0	0	927	33	2	42	8471044	19	0	1	0	2	2	4	0	0	0	174	0	3	3	21	8	
	SKINNY_JEANS	2	998	17	0	4	1	5	138	64	4	0	3531269	45	92	900378382	9	10	0	19	13	76	0	4	3	721	8	3	3	21	8	0	0	
	BOYFRIEND	1	176	1	0	1	3	1	20	23	1	0	242	24	4	9	211070581	3	0	0	8	0	15	0	1	0	40	0	0	0	0	0	0	0
	MINI_SKIRT	0	20	3	11	52	9	20	474	15	2	3	14	19	28	4	0	15	4	174893	3	247	2	1	5	132	0	5	0	0	0	0	0	0
	SPORTS_LEGGINGS	0	13	1	97	102	5	27	18	32	25	20	11	8	15	4	1	10	1	883127	72	2451	14	12	33	307	1	5	2	0	0	0	0	0
PENCIL_SKIRT	0	0	0	11	11	0	0	0	2	1	8	1	0	0	0	0	0	2	149	19	200	0	1	11	36	0	0	0	0	0	0	0	0	
SHORTS	0	25	10	182	694	22	294	41	55	49	71	22	16	50	7	5	10	3	1982292	687411	9	16	1761882	5	3	3	0	0	0	0	0	0	0	
DENIM_JACKET_CLASSIC	1	7	0	0	1	1	5	31	0	0	1	22	0	0	6	2	9	1	3	37	0	51	894	16	0	1	1	6	0	0	0	0	0	
SWEATER_DRESS	5	4	0	0	1	1	2	4	0	1	0	12	4	1	1	1	14	0	3	1	0	2	1	400	0	1	0	5	0	0	0	0	0	
BOOT_CUT	0	0	1	9	28	0	8	2	5	1	11	0	0	0	2	0	1	0	5	65	9	292	0	0	36	182	1	0	0	0	0	0	0	
STRAIGHT_JEANS	0	9	3	246	530	7	131	14	24	64	40	8	8	19	0	0	3	0	85	253	201819	0	9	852401	7	2	1	0	0	0	0	0	0	
FLARED_JEANS	0	7	0	1	0	1	108	17	3	0	5	2	2	0	0	1	13	1	1	0	0	20	1	0	0	6	618	1	0	0	0	0	0	
ACCESSORIES	0	94	1	0	1	1	4	19	8	0	0	122359	8	32	90	829	19	3	5	0	7	4	15	0	4	1	1704	3	2	2	2	2	0	
WAIST_BELTS	0	7	0	0	1	0	0	1	4	0	0	6	1	0	0	0	12	0	0	2	1	3	0	2	0	1	0	7	11	0	0	0	0	
T_SHIRT_DRESS	2	2	2	65	0	0	0	0	1	0	0	0	11	5	0	1	2	6	1	0	3	1	3	2	11	0	0	0	2	0	71	0	0	
		CHINOS	CLASSIC_SHORTS	ALL_JUMPSUITS	JERSEY_DRESS	SPORTS_SHOES	SHIFT_DRESSES	BELTS	OVERALLS	SKINNY	SLIM_LEG	STRAIGHT_LEG	JEANS	SHIRT_DRESSES	CARGO_PANTS	SHEATH_DRESS	STRAIGHT_LEG_TROUSERS	SKINNY_JEANS	BOYFRIEND	MINI_SKIRT	SPORTS_LEGGINGS	PENCIL_SKIRT	SHORTS	DENIM_JACKET_CLASSIC	SWEATER_DRESS	BOOT_CUT	STRAIGHT_JEANS	FLARED_JEANS	ACCESSORIES	WAIST_BELTS	T_SHIRT_DRESS			



PER CLASS ACCURACIES – LARGE SCALE (234K)

- Problem of different granularity of provided ground truth
- Parent/Child nodes used interchangeably
 - Misclassification of child as parent is not wrong
 - Model does not consider hierarchy



CONCLUSIONS

- Despite large dataset and reduced number of classes
 - **Pretrained models outperform** from-scratch training
 - Product classification – 79.1%
 - Gender prediction – 88.0%
- **Custom small model** enough to learn **binary task**
 - person/product classification – 91.07%
- **Preprocessing** of **ground-truth** required
 - Flatten hierarchy, remove ambiguities and overlaps
 - Use **hierarchical CNNs**
 - Use **attention** (person images)

THANK YOU!

Alexander Schindler, 08.06.2018

