

Learning Task-Specific Similarity

by

Gregory Shakhnarovich

Submitted to the Department of Electrical Engineering and Computer
Science

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2005

© Gregory Shakhnarovich, MMV. All rights reserved.

The author hereby grants to MIT permission to reproduce and
distribute publicly paper and electronic copies of this thesis document
in whole or in part.

Author
Department of Electrical Engineering and Computer Science
September 30, 2005

Certified by.....
Trevor J. Darrell
Associate Professor
Thesis Supervisor

Accepted by.....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Learning Task-Specific Similarity

by

Gregory Shakhnarovich

Submitted to the Department of Electrical Engineering and Computer Science
on September 30, 2005, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering and Computer Science

Abstract

The right measure of similarity between examples is important in many areas of computer science. In particular it is a critical component in example-based learning methods. Similarity is commonly defined in terms of a conventional distance function, but such a definition does not necessarily capture the inherent meaning of similarity, which tends to depend on the underlying task. We develop an algorithmic approach to learning similarity from examples of what objects are deemed similar according to the task-specific notion of similarity at hand, as well as optional negative examples. Our learning algorithm constructs, in a greedy fashion, an encoding of the data. This encoding can be seen as an embedding into a space, where a weighted Hamming distance is correlated with the unknown similarity. This allows us to predict when two previously unseen examples are similar and, importantly, to efficiently search a very large database for examples similar to a query.

This approach is tested on a set of standard machine learning benchmark problems. The model of similarity learned with our algorithm provides an improvement over standard example-based classification and regression. We also apply this framework to problems in computer vision: articulated pose estimation of humans from single images, articulated tracking in video, and matching image regions subject to generic visual similarity.

Thesis Supervisor: Trevor J. Darrell

Title: Associate Professor

Acknowledgments

I would like to thank my advisor, Trevor Darrell, for his influential role in this thesis and in my graduate career. Being Trevor's student has been fun in many ways; he has struck the perfect balance in his role as an advisor. On the one hand, he has given me a great deal of independence in pursuing my ideas and lots of encouragement. On the other hand he has provided thorough advice (on things academic and not) and, at times, challenging the nonsense I would produce. Not less importantly, Trevor has succeeded in building a really great research group, where things go smoothly professionally and socially.

Bill Freeman, Mike Collins and Shimon Ullman have been great thesis readers, and their careful and challenging feedback has made the thesis much better than it would otherwise be. I am also thankful for their patience with my ever shifting deadlines. My gratitude also goes to Tommi Jaakkola for many helpful conversations about learning, and for the enriching experience of TA-ing his machine learning course, and to Piotr Indyk, whose work has inspired much of the work described in the thesis and who was always willing to meet and talk. John Fisher has been an excellent colleague and friend, and I thank him for his appreciation of my Soviet jokes (one in particular).

My work and thinking has also been benefited by a number of collaborations outside of MIT. A summer internship at MERL with Paul Viola taught me many things about vision, learning and scientific intuition. Paul has contributed some of the ideas in the core of this thesis, and has collaborated with Trevor and me on the work presented in Chapter 4. I have also benefited from collaboration with Baback Moghaddam, Liu Ren, Jessica Hodgins, Paul Viola and Hanspeter Pfister; Liu led much of the effort in the last stages of the project described in Section 4.5.

During the five years I have spent at MIT, I have learned a great deal from courses and talks, but probably more so from the many seemingly random conversations with fellow students, staff and faculty. Naturally, the most frequent victims have been my officemates. I am very grateful to Leonid Taycher, Kristen Grauman and Ariadna Quattoni for many interesting conversations, for being kind and helpful in all matters, and for humoring me by listening to my ramblings and tolerating the (often bad) jokes. The other members of Trevor's research group have made my time here more productive and enjoyable. I would like to especially mention Mario Christoudias, David Demirdjian, Louis-Philippe Morency, Ali Rahimi, Kate Saenko

and Mike Siracusa who were always willing to talk about research, philosophy, politics and food. The work described in Section 5.3 is a collaborative effort with David, Leonid and Kristen. Many other people at MIT have been a privilege to know. I would especially like to thank Biswajit Bose, Polina Golland, Alex Ihler, Kevin Murphy, Bryan Russell, Erik Sudderth, Marshall Tappen, Kinh Tieu, Antonio Torralba and Lior Wolf for many useful and fun conversations. I would also like to acknowledge Rahul Sukhtankar, Yan Ke and Derek Hoiem from CMU for helpful conversations and for bringing to my attention the observation leading to the semi-supervised setup in Section 3.2.4.

Last but not least I would like to thank Misha, Olga, Gabi and Dorel who have provided the emotional backup (and occasionally nutrition and shelter) one needs in the journey through graduate school. My very special thanks of course go to my mother Lena, who I hope will be proud. Karen, nothing I could write here about how grateful I am to you for everything would be expressive enough!

Contents

1	Introduction	17
1.1	Modeling equivalence	18
1.1.1	Other notions of similarity	18
1.1.2	Example-based methods	19
1.1.3	Why learn similarity?	20
1.2	Learning embeddings that reflect similarity	21
1.2.1	Motivation: hashing and boosting	23
1.2.2	Similarity sensitive coding	23
1.2.3	Boosting the embedding bits	24
1.2.4	BoostPro: boosting optimized projections	25
1.2.5	Relationship to other similarity learning methods	25
1.3	Applications in computer vision	27
1.3.1	Levels of visual similarity	27
1.3.2	Example-based pose estimation	28
1.3.3	Learning visual similarity of image regions	29
1.4	Thesis organization	29
2	Background	31
2.1	Example-based classification	31
2.1.1	Properties of <i>KNN</i> classifiers	31
2.1.2	Approximate nearest neighbors	32
2.1.3	Near versus nearest	33
2.1.4	Evaluation of retrieval accuracy	34
2.2	Example-based regression	35
2.2.1	Regression-induced similarity	36
2.3	Learning Distances and Similarity	38
2.3.1	Metric learning	38
2.3.2	Similarity as classification	39
2.3.3	Embeddings and mappings	40
2.4	Algorithms for search and retrieval	41
2.4.1	kd-trees	41
2.4.2	Locality sensitive hashing	42
2.5	Summary	45

3	Learning embeddings that reflect similarity	47
3.1	Preliminaries	48
3.1.1	Threshold evaluation procedure	48
3.2	Similarity sensitive coding	49
3.2.1	Benchmark data sets	51
3.2.2	Performance and analysis of SSC	53
3.2.3	The coding perspective on SSC	56
3.2.4	Semi-supervised learning	57
3.2.5	Limitations of SSC	59
3.3	Ensemble embedding with AdaBoost	60
3.3.1	Boosting	61
3.3.2	Supervised boosted SSC	62
3.3.3	Boosting in a semi-supervised setup	64
3.4	BoostPro: boosting general projections	65
3.4.1	Embedding with generic projections	66
3.4.2	The weak learner of projections	67
3.4.3	Results	69
3.5	Discussion	79
4	Articulated Pose Estimation	81
4.1	The problem domain	81
4.2	Background on pose estimation	83
4.3	Example-based pose estimation	84
4.3.1	Pose-sensitive similarity	84
4.3.2	Image representation	86
4.3.3	Obtaining labeled data	87
4.4	Estimating upper body pose	87
4.4.1	Training data	88
4.4.2	The learning setup	88
4.4.3	Results	89
4.5	Estimating full body pose	92
4.5.1	Training data	92
4.5.2	Learning setup and results	92
4.6	Discussion	93
5	Articulated Tracking	95
5.1	Articulated tracking	95
5.1.1	Probabilistic tracking framework	96
5.1.2	Models of dynamics	96
5.1.3	Likelihood and similarity	97
5.2	Case I: Motion-driven animation	98
5.2.1	The setup and training data	98
5.2.2	Two-stage architecture	99
5.2.3	Performance	106
5.3	Case II: General pose tracking with likelihood modes	109

5.3.1	Pose similarity as likelihood sampling	109
5.3.2	Tracking with likelihood modes	109
5.3.3	Implementation and performance	110
5.4	Discussion	111
6	Learning Image Patch Similarity	115
6.1	Background	116
6.1.1	Patch similarity measures	116
6.1.2	Interest operators	118
6.2	Defining and labeling visual similarity	118
6.3	Patch descriptors	119
6.3.1	Sparse overcomplete code	121
6.3.2	SIFT	122
6.4	Experiments	123
6.4.1	Collecting descriptors	124
6.4.2	Embedding the descriptors for similarity	125
6.5	Discussion	129
7	Conclusions	133
7.1	Summary of thesis contributions	133
7.1.1	Learning algorithms	134
7.1.2	Example-based pose estimation	135
7.1.3	Articulated tracking	135
7.1.4	Patch similarity	135
7.2	Direction for future work	136

List of Figures

1-1	Task-specific similarity: toy 2d illustration.	22
1-2	Illustration of embeddings obtained with the learning algorithms. . .	26
1-3	Example-based pose estimation: a cartoon	28
2-1	Disambiguation by label clustering	37
2-2	Regression-induced similarity	38
2-3	Illustration of LSH lookup	44
3-1	Threshold evaluation algorithm	50
3-2	Distribution of gap	55
3-3	Distribution of threshold TP/FP rates	56
3-4	Covariances of SSC bits	58
3-5	Angle and norm similarity in 2D	66
3-6	Results on Auto-MPG	71
3-7	Results on Machine CPU	71
3-8	Results on Boston Housing	72
3-9	Results on Abalone	72
3-10	Results on US Census	73
3-11	Results on Letter	73
3-12	Results on Isolet	74
3-13	Weak classifiers for synthetic 2D data	74
3-14	Similarity regions for synthetic 2D data	75
3-15	Results of semi-supervised BoostPro on synthetic data	75
3-16	Results of semi-supervised BoostPro on UCI data	76
3-17	Effect of similarity rate on semi-supervised BOOSTPRO	78
4-1	Articulate model of a human body	82
4-2	Edge direction histograms	86
4-3	Example training images for upper body pose estimation	88
4-4	Testing upper body pose estimation on real images-I	90
4-5	Testing upper body pose estimation on real images-II	91
4-6	Example training images for full body pose estimation	92
4-7	Retrieval results with BoostPro, synthetic input.	93
4-8	Retrieval results with BoostPro, real input.	94
5-1	Embedding projections for for yaw similarity	101

5-2	Training examples for yaw similarity	102
5-3	Test error for yaw and pose similarity classifiers	103
5-4	Performance of yaw similarity detector	104
5-5	Training examples for pose similarity	105
5-6	Retrieval error of similarity embedding vs. Hu moments	107
5-7	Comparison of pose retrieval results	108
5-8	Estimation loop in ELMO	110
5-9	Comparative error analysis of ELMO	112
5-10	Comparative tracking results with ELMO	113
5-11	Tracking results extracted from the dance sequence	114
5-12	Tracking results extracted from the whiteboard sequence	114
6-1	Examples of similar patches	118
6-2	Descriptors for example patches	120
6-3	Examples of whitened images	123
6-4	Example basis functions for sparse overcomplete code	125
6-5	ROC curves with sparse overcomplete codes	126
6-6	ROC curves with SIFT descriptors	127
6-7	Example projections for sparse code embedding	128
6-8	Embeddings of the descriptors for example patches	131
7-1	Proposed feature hierarchy for object representation	138

List of Tables

3.1	Summary of the data sets used in the evaluation.	53
3.2	Encoding lengths	57
3.3	Regression accuracy on UCI/Delve data, MAE	69
3.4	Classification accuracy with SSC on UCI/Delve data sets	69
3.5	Regression accuracy on UCI/Delve data, MSE	70
3.6	Best of other published results	70
3.7	BOOSTPROembedding length, real data sets	76
4.1	Pose estimation accuracy on synthetic images	89
6.1	Area under ROC for similarity measures compared in our evaluation.	126

List of Algorithms

1	<i>K</i> nearest neighbors classification	32
2	<i>R</i> -neighbor classification	33
3	LSH construction	43
4	Projection threshold evaluation	51
5	Basic similarity sensitive coding	52
6	Semi-supervised threshold evaluation	60
7	Boosted SSC	63

