Tombone's Computer Vision Blog

Deep Learning, Computer Vision, and the algorithms that are shaping the future of Artificial Intelligence.

Tuesday, January 20, 2015

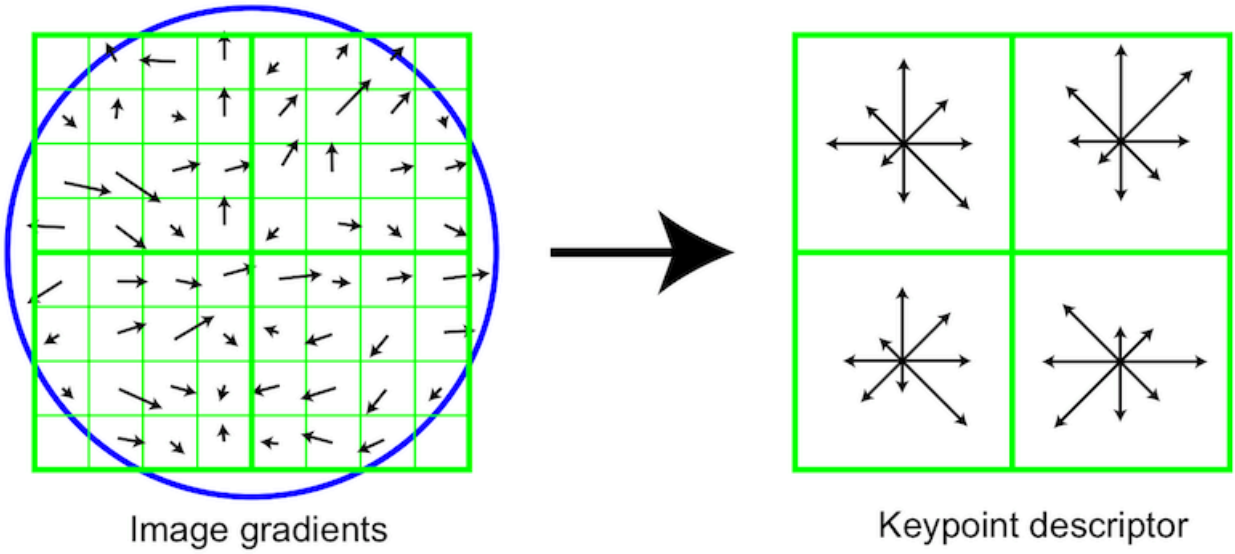
From feature descriptors to deep learning: 20 years of computer vision

We all know that deep convolutional neural networks have produced some stellar results on object detection and recognition benchmarks in the past two years (2012-2014), so you might wonder: *what did the earlier object recognition techniques look like*? *How do the designs of earlier recognition systems relate to the modern multi-layer convolution-based framework*?

Let's take a look at some of the big ideas in Computer Vision from the last 20 years.

The rise of the local feature descriptors: ~1995 to ~2000

When **SIFT** (an acronym for Scale Invariant Feature Transform) was introduced by **David Lowe** in 1999, the world of computer vision research changed almost overnight. It was robust solution to the problem of comparing image patches. Before SIFT entered the game, people were just using SSD (sum of squared distances) to compare patches and not giving it much thought.



The SIFT recipe: gradient orientations, normalization tricks

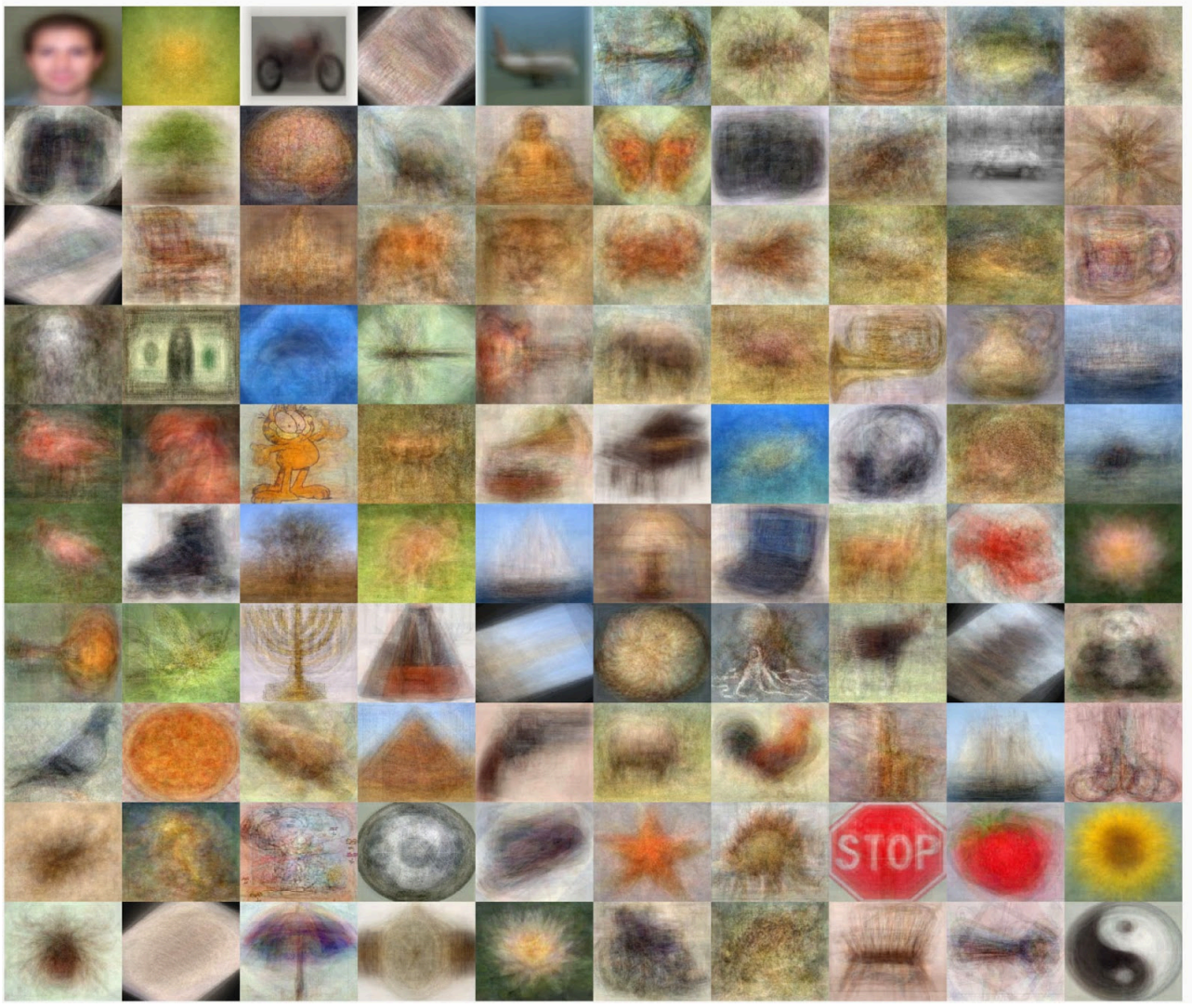
SIFT is something called a local feature descriptor -- it is one of those research findings which is the result of one ambitious man hackplaying with pixels for more than a decade. Lowe and the University of British Columbia got a patent on SIFT and *Lowe released a nice compiled binary of his very own SIFT implementation for researchers to use in their work*. SIFT allows a point inside an RGB imagine to be represented robustly by a low dimensional vector. When you take multiple images of the same physical object while rotating the camera, the SIFT descriptors of corresponding points are very similar in their 128-D space. At first glance it seems silly that you need to do something as complex as SIFT, but believe me: just because you, a human, can look at two image patches and quickly "understand" that they belong to the same physical point, this is not the same for machines. SIFT had massive implications for the geometric side of computer vision (stereo, Structure from Motion, etc) and later became the basis for the popular Bag of Words model

for object recognition.

Seeing a technique like SIFT dramatically outperform an alternative method like Sum-of-Squared-Distances (SSD) Image Patch Matching firsthand is an important step in every aspiring vision scientist's career. And SIFT isn't just a vector of filter bank responses, the binning and normalization steps are very important. It is also worthwhile noting that while SIFT was initially (in its published form) applied to the output of an interest point detector, later it was found that the interest point detection step was not important in categorization problems. For categorization, researchers eventually moved towards vector quantized SIFT applied densely across an image.

I should also mention that other descriptors such as **Spin Images** (see my 2009 blog post on spin images) came out a little bit earlier than SIFT, but because Spin Images were solely applicable to 2.5D data, this feature's impact wasn't as great as that of SIFT.

The modern dataset (aka the hardening of vision as science): ~2000 to ~2005
Homography estimation, ground-plane estimation, robotic vision, SfM, and all other geometric problems in vision greatly benefited from robust image features such as SIFT. But towards the end of the 1990s, it was clear that *the internet was the next big thing*. Images were going online. Datasets were being created. And no longer was the current generation solely interested in structure recovery (aka geometric) problems. This was the beginning of the large-scale dataset era with Caltech-101 slowly gaining popularity and categorization research on the rise. No longer were researchers evaluating their own algorithms on their own in-house datasets -- we now had a more objective and standard way to determine if yours is bigger than mine. Even though Caltech-101 is considered outdated by 2015 standards, it is fair to think of this dataset as the Grandfather of the more modern ImageNet dataset. Thanks Fei-Fei Li.



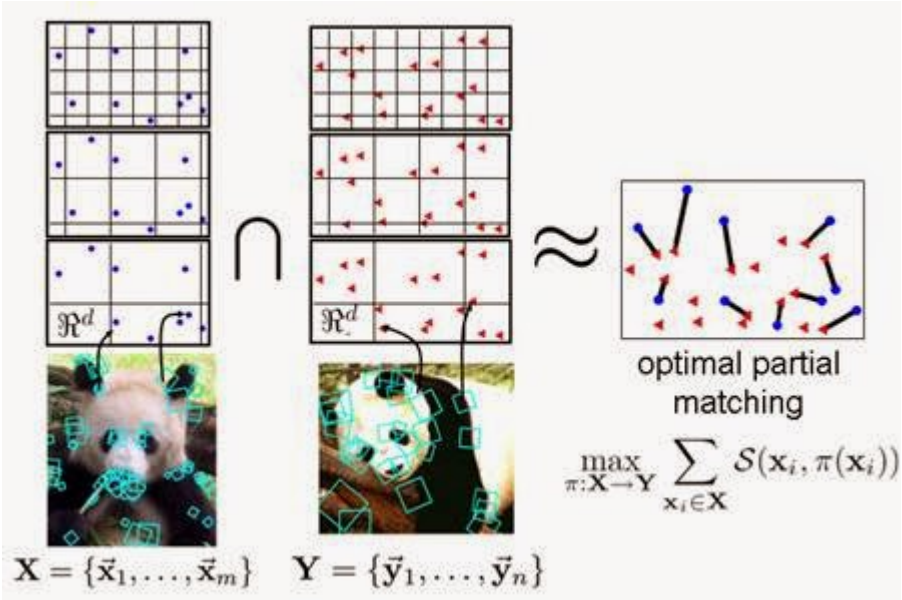
Category-based datasets: the infamous Caltech-101 TorralbaArt image

Bins, Grids, and Visual Words (aka Machine Learning meets descriptors): ~2000 to ~2005
After the community shifted towards more ambitious object recognition problems and away from geometry recovery problems, we had a flurry of research in Bag of Words, Spatial Pyramids, Vector Quantization, as well as machine learning tools used in any and all stages of the computer vision pipeline. Raw SIFT was great for wide-baseline stereo, but it wasn't powerful enough to provide matches between two distinct object instances from the same visual object category. What was needed was a way to encode the following ideas: object parts can deform relative to each other and some image patches can be missing. Overall, a much more *statistical way to characterize objects was needed*.

Visual Words were introduced by Josef Sivic and Andrew Zisserman in approximately 2003 and this was a clever way of taking algorithms from large-scale text matching and applying them to visual content. A visual dictionary can be obtained by performing unsupervised learning (basically just K-means) on SIFT descriptors which maps these 128-D real-valued vectors into integers (which are cluster center assignments). A histogram of these visual words is a fairly robust way to represent images. Variants of the Bag of Words model are still heavily utilized in vision research.

Josef Sivic's "Video Google": Matching Graffiti inside the Run Lola Run video

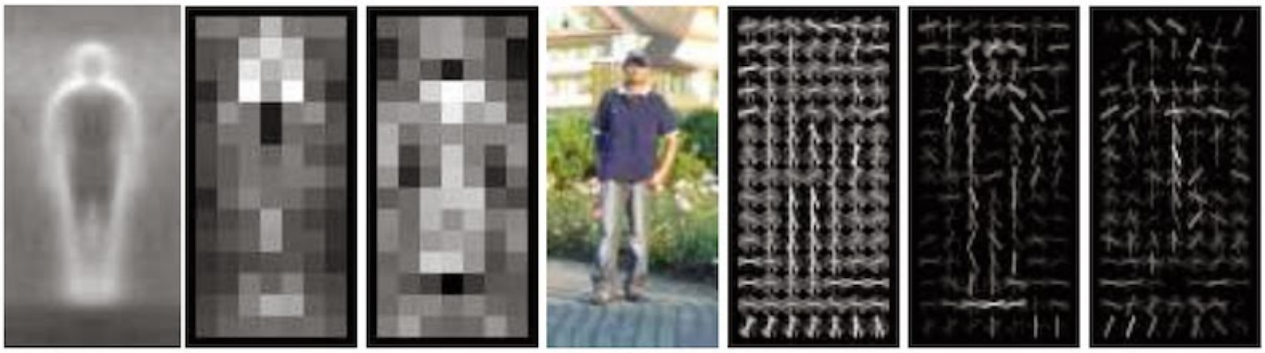
Another idea which was gaining traction at the time was the idea of using some sort of binning structure for matching objects. Caltech-101 images mostly contained objects, so these grids were initially placed around entire images, and later on they would be placed around object bounding boxes. Here is a picture from Kristen Grauman's famous Pyramid Match Kernel paper which introduced a powerful and hierarchical way of integrating spatial information into the image matching process.



Grauman's Pyramid Match Kernel for Improved Image Matching

At some point it was not clear whether researchers should focus on better features, better comparison metrics, or better learning. In the mid 2000s it wasn't clear if young PhD students should spend more time concocting new descriptors or kernelizing their support vector machines to death.

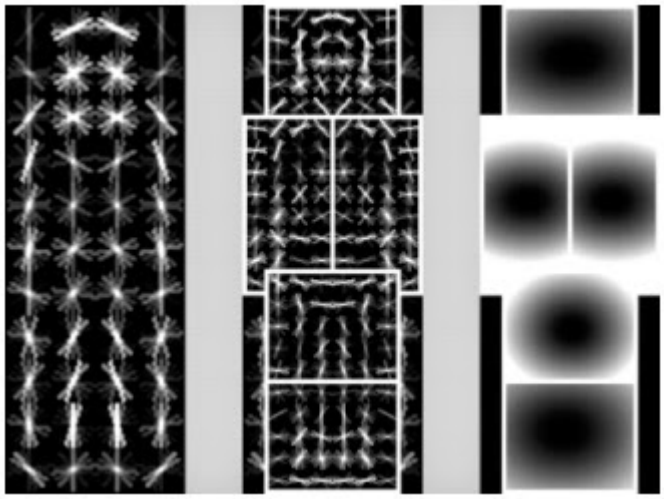
Object Templates (aka the reign of HOG and DPM): ~2005 to ~2010
At around 2005, a young researcher named Navneet Dalal showed the world just what can be done with his own new badass feature descriptor, HOG. (It is sometimes written as HoG, but because it is an acronym for "Histogram of Oriented Gradients" it should really be HOG. The confusion must have came from an earlier approach called DoG which stood for Difference of Gaussian, in which case the "o" should definitely be lower case.)



Navneet Dalal's HOG Descriptor

HOG came at the time when everybody was applying spatial binning to bags of words, using multiple layers of learning, and making their systems overly complicated. Dalal's ingenious descriptor was actually quite simple. The seminal HOG paper was published in 2005 by Navneet and his PhD advisor, Bill Triggs. Triggs got his fame from earlier work on geometric vision, and Dr. Dalal got his fame from his newly found descriptor. HOG was initially applied to the problem of pedestrian detection, and one of the reasons it because so popular was that the machine learning tool used on top of HOG was quite simple and well understood, it was the linear Support Vector Machine.

I should point out that in 2008, a follow-up paper on object detection, which introduced a technique called the Deformable Parts-based Model (or DPM as we vision guys call it), helped reinforce the popularity and strength of the HOG technique. I personally jumped on the HOG bandwagon in about 2008. My first few years as a grad student (2005-2008) I was hackplaying with my own vector quantized filter bank responses, and definitely developed some strong intuition regarding features. In the end I realized that my own features were only "okay," and because I was applying them to the outputs of image segmentation algorithms they were extremely slow. Once I started using HOG, it didn't take me long to realize there was no going back to custom, slow, features. Once I started using a multiscale feature pyramid with a slightly improved version of HOG introduced by master hackers such as Ramanan and Felzenszwalb, I was processing images at 100x the speed of multiple segmentations + custom features (my earlier work).



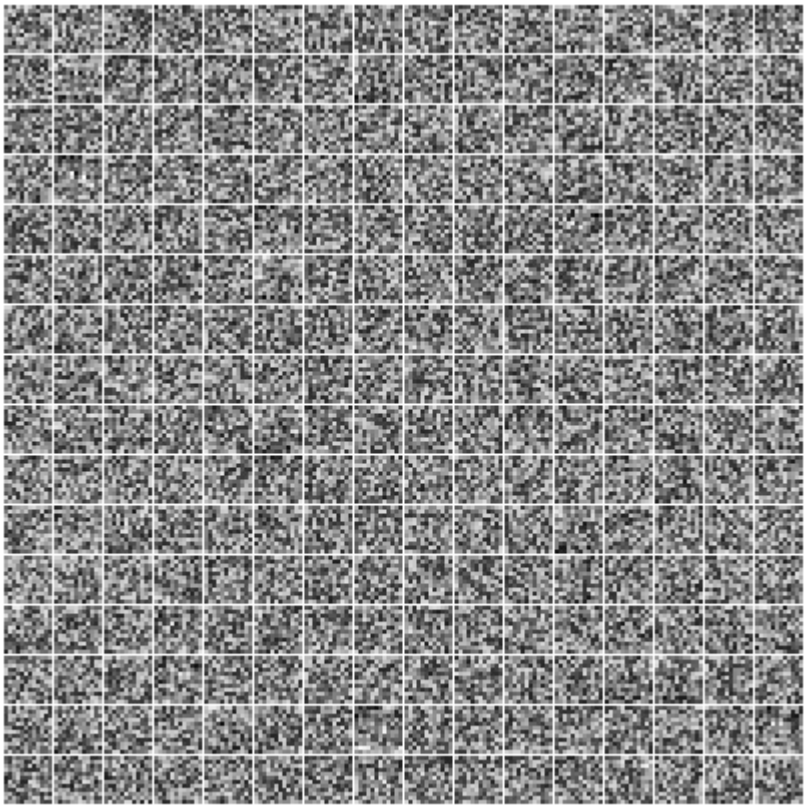
The infamous Deformable Part-based Model (for a Person)

DPM was the reigning champ on the PASCAL VOC challenge, and one of the reasons why it became so popular was *the excellent MATLAB/C++ implementation by Ramanan and Felzenszwalb*. I still know many researchers who never fully acknowledged what releasing such great code really meant for the fresh generation of incoming PhD students, but at some point it seems like everybody was modifying the DPM codebase for their own CVPR attempts. Too many incoming students were lacking solid software engineering skills and giving them the DPM code was a surefire way to get some some experiments up and running. Personally, I never jumped on the parts-based methodology, but I did take apart the DPM codebase several times. However, when I put it back together, the Exemplar-SVM was the result.

Big data, Convolutional Neural Networks and the promise of Deep Learning: ~2010 to ~2015
Sometime around 2008, it was pretty clear that scientists were getting more and more comfortable with large datasets. It wasn't just the rise of "Cloud Computing" and "Big Data," it was the rise of the data scientists. Hacking on equations by morning, developing a prototype during lunch, deploying large scale computations in the evening, and integrating the findings into a production system by sunset. I spent two summers at Google Research, I saw lots of guys who had made their fame as vision hackers. But they weren't just writing "academic" papers at Google -- sharding datasets with one hand, compiling results for their managers, writing Borg scripts in their sleep, and piping results into gnuplot (because Jedis don't need GUIs?). It was pretty clear that big data, and a DevOps mentality was here to stay, and the vision researcher of tomorrow would be quite comfortable with large datasets. No longer did you need one guy with a mathy PhD, one software engineer, one manager, and one tester. Plenty of guys who could do all of those jobs.

Deep Learning: 1980s - 2015
2014 was definitely a big year for Deep Learning. What's interesting about Deep Learning is that it is a very old technique. What we're seeing now is essentially the Neural Network 2.0 revolution -- but this time around, there's we're 20 years ahead R&D-wise and our computers are orders of magnitude faster. And what's funny is that the same guys that were championing such techniques in the early 90s were the same guys we were laughing at in the late 90s (because clearly convex methods were superior to the magical NN learning-rate knobs). I guess they really had the last laugh because eventually these relentless neural network gurus became the same guys we now all look up to. **Geoffrey Hinton, Yann LeCun, Andrew Ng, and Yeshua Bengio are the 4 Titans of Deep Learning.** By now, just about everybody has jumped ship to become a champion of Deep Learning.

But with Google, Facebook, Baidu, and a multitude of little startups riding the Deep Learning wave, **who will rise to the top as the master of artificial intelligence?**



Yann's Deep Learning Page

How to today's deep learning systems resemble the recognition systems of yesteryear?
Multiscale convolutional neural networks aren't that much different than the feature-based systems of the past. The first level neurons in deep learning systems learn to utilize gradients in a way that is similar to hand-crafted features such as SIFT and HOG. Objects used to be found in a sliding-window fashion, but now it is easier and sexier to think of this operation as convolving an image with a filter. Some of the best detection systems used to use multiple linear SVMs, combined in some ad-hoc way, and now we are essentially using even more of such linear decision boundaries. Deep learning systems can be thought of a multiple stages of applying linear operators and piping them through a non-linear activation function, but deep learning is more similar to a clever combination of linear SVMs than a memory-ish Kernel-based learning system.

Features these days aren't engineered by hand. However, architectures of Deep systems are still being designed manually -- and it looks like the experts are the best at this task. The operations on the inside of both classic and modern recognition systems are still very much the same. You still need to be clever to play in the game, but *now you need a big computer*. There's still lot of room for improvement, so I encourage all of you to be creative in your research.

Research-wise, it never hurts to know where we have been before so that we can better plan for our journey ahead. I hope you enjoyed this brief history lesson and the next time you look for insights in your research, don't be afraid to look back.

To learn more about computer vision techniques:
SIFT article on Wikipedia
Bag of Words article on Wikipedia
HOG article on Wikipedia
Deformable Part-based Model Homepage
Pyramid Match Kernel Homepage
"Video Google" Image Retrieval System

Some Computer Vision datasets:
Caltech-101 Dataset

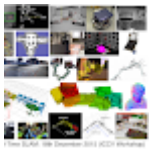
ImageNet Dataset

To learn about the people mentioned in this article:
Kristen Grauman (creator of Pyramid Match Kernel, Prof at Univ of Texas)
Bill Triggs's (co-creator of HOG, Researcher at INRIA)
Navneet Dalal (co-creator of HOG, now at Google)
Yann LeCun (one of the Titans of Deep Learning, at NYU and Facebook)
Geoffrey Hinton (one of the Titans of Deep Learning, at Univ of Toronto and Google)
Andrew Ng (leading the Deep Learning effort at Baidu, Prof at Stanford)
Yoshua Bengio (one of the Titans of Deep Learning, Prof at U Montreal)
Deva Ramanan (one of the creators of DPM, Prof at UC Irvine)
Pedro Felzenszwalb (one of the creators of DPM, Prof at Brown)
Fei-fei Li (Caltech101 and ImageNet, Prof at Stanford)
Josef Sivic (Video Google and Visual Words, Researcher at INRIA/ENS)
Andrew Zisserman (Geometry-based methods in vision, Prof at Oxford)
Andrew E. Johnson (SPIN Images creator, Researcher at JPL)
Martial Hebert (Geometry-based methods in vision, Prof at CMU)

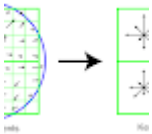
Posted by Tomasz Malisiewicz at Tuesday, January 20, 2015
Email ThisBlogThis!Share to XShare to FacebookShare to Pinterest
Labels: bengio, big data, convolution, dalal, deep learning, detection, feature engineering, features, google, hinton, HOG, learning, lecun, lowe, machine learning, object recognition, sift, vision
Newer Post Older Post Home
Subscribe to: Post Comments (Atom)

Popular Posts

- Deep Learning vs Machine Learning vs Pattern Recognition
  Lets take a close look at three related terms (Deep Learning vs Machine Learning vs Pattern Recognition), and see how they relate to some o...



- The Future of Real-Time SLAM and Deep Learning vs SLAM
  Last month's International Conference of Computer Vision (ICCV) was full of Deep Learning techniques, but before we declare an all-out...



- From feature descriptors to deep learning: 20 years of computer vision
  We all know that deep convolutional neural networks have produced some stellar results on object detection and recognition benchmarks in th...

- Deep Learning Trends @ ICLR 2016
  Started by the youngest members of the Deep Learning Mafia [1], namely Yann LeCun and Yoshua Bengio , the ICLR conference is quickly becom...

- ICCV 2015: Twenty one hottest research papers
  "Geometry vs Recognition" becomes ConvNet-for-X Computer Vision used to be cleanly separated into two schools: geometry and rec...

- Deep Learning vs Probabilistic Graphical Models vs Logic
  Today, let's take a look at three paradigms that have shaped the field of Artificial Intelligence in the last 50 years: Logic , Probab...

- Can a person-specific face recognition algorithm be used to determine a person's race?
  It's a valid question: can a person-specific face recognition algorithm be used to determine a person's race? I trained two separa...

Recent Posts

- Computer Vision and Visual SLAM vs. AI Agents

With all the recent advancements in end-to-end deep learning, it is now ... read more
Nov 19 2019

- DeepFakes: AI-powered deception machines

Driven by computer vision and deep learning techniques, a new wave of imaging ... read more
May 16 2018

- Nuts and Bolts of Building Deep Learning Applications: Ng @ NIPS2016

You might go to a cutting-edge machine learning research conference like NIPS ... read more
Dec 16 2016

- Making Deep Networks Probabilistic via Test-time Dropout

In Quantum Mechanics, Heisenberg's Uncertainty Principle states that there is a ... read more
Jun 17 2016

- Deep Learning Trends @ ICLR 2016

Started by the youngest members of the Deep Learning Mafia [1], ... read more
Jun 01 2016

Recent Posts Widget

About Me

Tomasz Malisiewicz
View my complete profile

Links

- Tomasz @ MIT Research Homepage
- Tomasz @ Google Scholar Citations
- Tomasz @ Github Open-Source Code

Blog Archive

- ► October (1)
- ► September (1)
- ► July (1)
- ► June (3)
- ► April (2)

- ► 2012 (11)
  - ► July (1)
  - ► June (4)
  - ► May (1)
  - ► April (2)
  - ► March (1)
  - ► January (2)
- ► 2011 (31)
  - ► December (4)
  - ► November (3)
  - ► October (3)
  - ► September (3)
  - ► August (5)
  - ► July (3)
  - ► June (2)
  - ► April (1)
  - ► March (5)
  - ► January (2)
- ► 2010 (24)
  - ► December (1)
  - ► November (2)
  - ► August (2)

- ► June (5)
- ► May (2)
- ► April (3)
- ► March (3)
- ► February (1)
- ► January (5)

- ► 2009 (29)
  - ► December (2)
  - ► November (4)
  - ► October (3)
  - ► September (1)
  - ► August (3)
  - ► July (3)
  - ► June (4)
  - ► March (5)
  - ► February (2)
  - ► January (2)
- ► 2008 (23)
  - ► December (2)
  - ► November (3)
  - ► October (1)
  - ► September (1)
  - ► August (1)
  - ► July (3)
  - ► June (3)
  - ► May (2)
  - ► April (3)

- ► March (2)
- ► February (2)

- ► 2007 (11)
  - ► September (1)
  - ► August (2)
  - ► July (1)
  - ► June (1)
  - ► April (1)
  - ► March (1)
  - ► February (1)
  - ► January (3)

- ► 2006 (58)
  - ► December (3)
  - ► November (1)
  - ► October (1)
  - ► September (4)
  - ► August (3)
  - ► July (1)
  - ► June (4)
  - ► May (5)
  - ► April (7)
  - ► March (8)
  - ► February (8)
  - ► January (13)

- ► 2005 (62)
  - ► December (12)
  - ► November (12)

- ► October (10)
- ► September (15)
- ► August (13)

Labels

- 3d recognition
- abhinav gupta
- antonio torralba
- artificial intelligence
- cognitive science
- computer vision
- cvpr
- deep learning
- entrepreneurship
- future directions
- graphical models
- iccv
- image understanding
- MATLAB
- MIT
- nips
- object recognition
- philosophy
- programming
- psychology
- scene understanding
- segmentation
- startups
- svm
- training
- visual memex
- VMX