# Computer Vision News

The magazine of the algorithm community

A publication by RSIP Vision

February 2018


Project Management by Ron Soferman:
Deep Learning: Thriving for More than 90%

Application:
UnifyID - Implicit Authentication

Review of Research Paper:
Deep Learning using Linear SVM

RSIP Vision:
Machine Vision for Robotics

We Tried for You:
TensorFlow-Slim

Spotlight News

Upcoming Events

Guest:
Prof. Emeritus Roy Davies

Fail of the Month:
Google Photos - Panorama

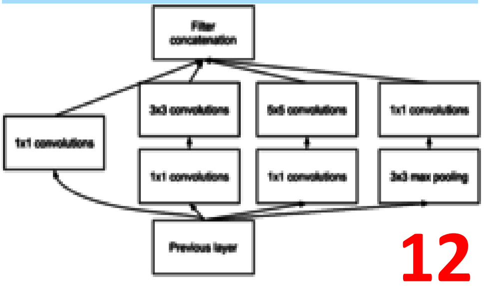
Challenge:
Learned Image Compression

## Women in Computer Vision: Michela Paganini

## Research
Deep Learning using Linear SVM

04

## We Tried for You
TensorFlow-Slim




12

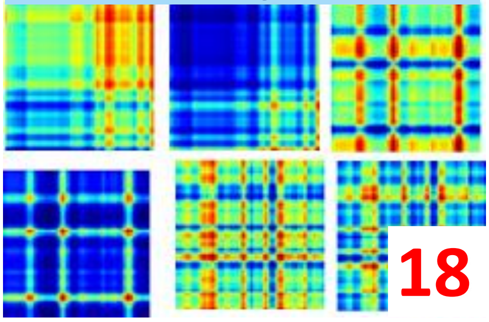
## Challenge - CLIC
Learned Image Compression

26

## Project
Machine Vision for Robots

08

## Applications
UnifyID



18

## Project Management
by Ron Soferman

27

## Spotlight News

10

## Guest
Prof. Emeritus Roy Davies

22

## Women in Computer Vision
Michela Paganini

28

Did you subscribe to Computer Vision News? It's free, click here!

**Computer Vision News**

Editor:
**Ralph Anzarouth**

Engineering Editor:
**Assaf Spanier**

Publisher:
**RSIP Vision**

Contact us

Give us feedback

Free subscription

Read previous magazines



Dear reader,

As every month for the last two years, we present a brand new issue of Computer Vision News. As usual, the editors have prepared a brilliant set of articles spanning from machine vision to image processing, from real-world applications to programming tools for our readers.

The guest of the month is **Roy Davies, Emeritus Professor of Machine Vision** at the **University of London** and author of **Computer Vision**. While we are making place for the fifth edition of his book on the shelves of our libraries (with new content about **Machine Learning** and **Deep Learning Neural Networks**), Prof. Davies is kindly available to share with us his vision regarding science, academy, career and, of course, computer vision. Read the interview at page 22.

If you like curiosities, you will find in this issue for the first time a "**Fail of the Month**" section: as we all know, **image processing** is not always successful at first try and sometimes the results are hilarious. See for yourself at page 11 an instance where **Google Photos** believed it could stitch three vacation pictures together: indeed, the panorama is impressive in several ways...

Don't miss all our regular sections, in particular our exclusive technical articles: the Research of the Month and We Tried for You.

**Enjoy the reading!**

**Ralph Anzarouth**
Marketing Manager, **RSIP Vision**
Editor, **Computer Vision News**

# Deep Learning using Linear Support Vector Machines
## SVM or Softmax?

by Assaf Spanier


Every month, Computer Vision News reviews a research paper from our field. This month we have chosen to review **Deep Learning using Linear Support Vector Machines**. We are indebted to the author **Yichuan Tang** from the **Department of Computer Science, University of Toronto**, for allowing us to use images from the paper to illustrate this review. His work is here.

*"Evaluate an SVM loss function as an alternative to softmax"*

## Introduction:

Convolutional deep learning neural networks have greatly evolved in recent years, achieving state of the art results in a wide range of computer vision tasks. These achievements include (among others) classification, object detection, localization and segmentation. Most CNNs using fully connected layers and convolutional layers rely on the softmax function for estimating network loss.

The author of this paper evaluates an SVM loss function as an alternative to softmax.

His results show that replacing softmax with L2-SVM produces a gain when run on the popular MNIST, CIFAR-10 and Facial Expression datasets, indicating that softmax's status as the go-to solution among developers should be reconsidered.

To be able to make a comparison in the same setting between softmax and an SVM function, he needed a differentiable loss function - and chose L2-SVM (SVM squared). The author reproduced the model proposed in Zhong & Ghosh, 2000, but used the L2-SVM in place of the standard SVM (hinge loss).

| | Softmax | SVM |
|---|---|---|
| **The function** | $L_i = -\log\left(\frac{\exp(s_k)}{(\sum_j \exp(s_j))}\right)$<br><br>$s_k$ – Score of the k-th category | $L_i = \min_{w,\varepsilon_n} \frac{1}{2} w^T \cdot w + C \sum_{n=1}^{N} \varepsilon_n$<br><br>$s.t. w_t x_n \geq 1 - \varepsilon_n \ \forall n \ (\varepsilon_n > 0)$<br><br>$L_i = \frac{1}{2} w^T \cdot w + C \sum_{n=1}^{N} \max(1 - w^T x_n, 0)$ |

| | | |
|---|---|---|
| **Derivative** | $\frac{\partial L_i}{\partial s_k}$=-1+$\frac{\exp(s_k)}{(\sum_j \exp(s_j))}$ =-1+$\frac{\exp(s_k)}{(\sum_j \exp(s_j))}$ | $\frac{\partial L_i}{\partial h_k} = -2 \cdot w(\max(1 - w^T h_n, 0))$ |
| **Objectives** | Minimizes cross-entropy or maximizes the log-likelihood | Maximum margin between data points of different classes (one-vs-rest approach) |
| **Loss function property** | Gives a probability of correct classification (never 100%); as a consequence, the learning process could theoretically continue indefinitely | Returns 0 when correctly classified; doesn't differentiate between small-margin and large-margin correct classification |
| **Inter-class loss** | Cross entropy | one-vs-rest |

# Dataset:

The following datasets were used for evaluation:

1. MNIST - a benchmark for the handwritten **digit** 10-class classification task, including 60,000 **grayscale** training images and 10,000 test cases. For training Gaussian noise was added to the data.
2. CIFAR-10 - a benchmark dataset for the 10-class **object** classification task, including 50,000 (32 × 32 pixel) **colored** images for training and 10,000 for testing. For training, horizontal reflection and jitter were randomly applied to the data.
3. Facial Expression Recognition (classification) challenge at the ICML 2013 Representation Learning Workshop at the University of Montreal - the dataset consisted of 28,709 (48x48 pixel) images of **faces** with **7 expression-types** for training and 3,589 test images.

# Networks used:

## MNIST

The network used for MNIST consisted of 2 fully-connected hidden layers of 512 units each, followed by either the softmax or the L2-SVM loss function. The data was divided into 300 batches of 200 samples each. Linear decay from 0.1 to 0.0 was used. The L2 weight cost on the softmax was set to 0.001.

### CIFAR

The CNN used for CIFAR consisted of 2 convolutional layers: the first had 32 5×5 filters, the second had 64 5 × 5 filters, the first followed by a Relu activation layer and both followed by max pooling layers, downsampled by a factor of 2.

### Facial Expression Recognition challenge

For the Facial Expression the CNN used an image mirroring layer, similarity transformation layer, two convolutional filtering + pooling stages, followed by a fully connected layer with 3072 hidden units. Source code

## Results:

The author showed that a CNN network with L2-SVM loss function outperformed softmax on 2 popular benchmark classification datasets. Results are as follows:

### MNIST

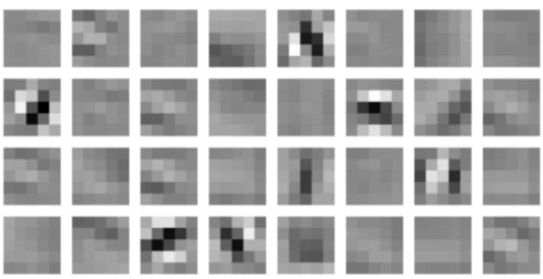
| | ConvNet+Softmax | ConvNet+SVM |
|---|---|---|
| Test error | 0.99% | 0.87% |

### CIFAR

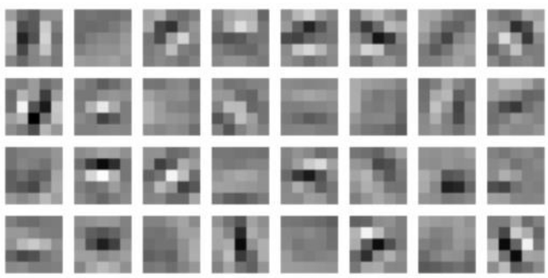
| | ConvNet+Softmax | ConvNet+SVM |
|---|---|---|
| Test error | 14.0% | 11.9% |

### Facial Expression

| | ConvNet+Softmax | ConvNet+SVM |
|---|---|---|
| Accuracy | 14.0% | 11.9% |



*Filters from convolutional net with softmax.*



*Filters from convolutional net with L2-SVM.*

## SVM trained conv net appears to have more textured filters!

To verify the gain in accuracy of the CNN with L2-SVM over the CNN with softmax was indeed due to the loss function and not to network optimization, the author checked what objective function score each of the two networks produced for the other loss function as well for the loss function it was optimizing for. The table below shows the networks did not produce good results for the **other** loss function - therefore the loss functions **should** be credited (blamed) for the difference in accuracy.

| | ConvNet+Softmax | ConvNet+SVM |
|---|---|---|
| Test error | 14.0% | 11.9% |
| Avg. cross entropy | 0.072 | 0.353 |
| Hinge loss squared | 213.2 | 0.313 |

Moreover, when the author took the weights of the CNN+L2-SVM that achieved the 11.9% error rate, and initialized a CNN+softmax with them, further training deteriorated the error rate towards 14%.

## Conclusion:

The author showed that a **CNN network** with L2-SVM loss function **outperformed softmax** on 2 popular benchmark classification datasets. This suggests developers should evaluate the **performance of softmax and SVM** before selecting which one to use, all the more so because switching between them is not complicated. Further evidence that SVM performs better than softmax may be found in other articles, including “DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition”, by Jeff Donahue et al.

Donahue’s team tested the classification capabilities of AlexNet layers 5, 6 and 7 (each on its own) on a variety of different databases (different from those on which AlexNet was originally trained) -- evaluating their performance using either softmax or SVM as the loss function -- and showed that at least in some cases **SVM was preferable**.

**A CNN network with L2-SVM loss function outperformed softmax on 2 popular benchmark classification datasets. This suggests that developers should evaluate the performance of softmax and SVM before selecting which one to use**

Project

Every month, Computer Vision News reviews a successful project. Our main purpose is to show how diverse image processing applications can be and how the different techniques contribute to solving technical challenges and physical difficulties. This month we review **RSIP Vision's approach to Machine Vision for Robots and our experience in Robotics projects**. RSIP Vision's engineers can assist you in countless application fields. **Get effective R&D consulting from our experts now!**

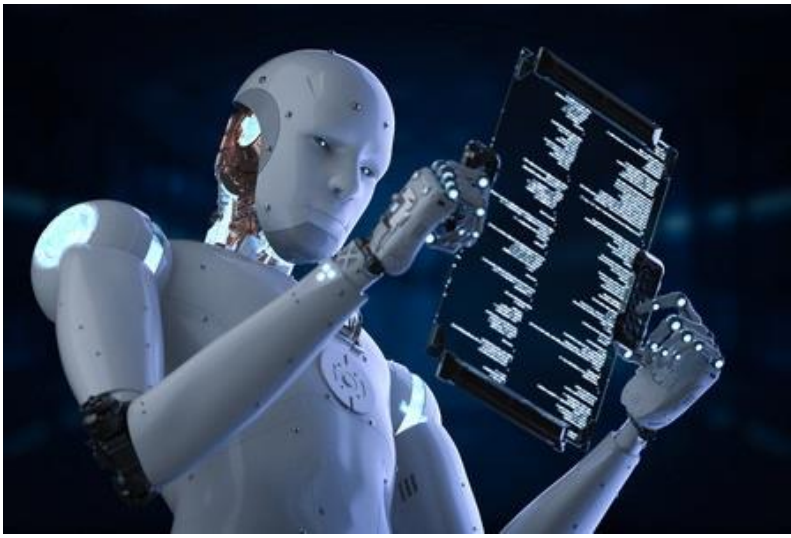
***"Programmed robots use machine vision to verify their action and they use machine vision to make sure that their mission is carried successfully"***

Robots performing human tasks in human environment are mostly built to resemble humans; robots doing other tasks, like car assembly, object picking, sorting and the like, generally take any form which is optimal to their function.

Another distinction we need to do is between pre-programmed robots in charge of repetitive tasks and robots that handle events, like navigation, picking, and so on. Programmed robots use machine vision to verify their action: they know what to do - in predefined sequence - and they use machine vision to make sure that their mission is carried successfully. An example in **industrial robotics** is robot welding, which includes a post-welding analysis phase during which a camera is used to verify that the work was executed correctly.

***"Robots use machine vision to build a dynamic map of the real world"***

Robots facing events, which cannot be programmed using some routine sequence, use **machine vision** to build a dynamic map of the real world, i.e. their surrounding environment. This map helps them avoid obstacles, navigate as needed, recognize objects which need to be picked, removed or sorted, and so on.



Another type of machine vision tasks is **OCR for robotics**: robots complete their vision with textual data appearing in their visual field.

Machine vision should be highly accurate, especially when the robot faces a changing environment, like an event. When conditions change (lighting, viewing angles, moving or changing objects) accuracy of machine vision is of crucial importance, to make sure that the robot can carry its task: navigating robots, like in a warehouse or in retail, need to move between people and/or other robots. Objects may not be oriented or located in the right position and since the robot is using its arms to pick and place, the need for precise guidance of the arm into the location is key for the success of the task. All the more so for robots doing surgery or other medical tasks.

***"Accuracy of Machine Vision is of crucial importance, to make sure that the robot can carry its task "***

This imposes a great challenge, because classic machine vision algorithms cannot be programmed to handle such a wide range of possible events: changing combination of parts, size and orientation, changing light conditions and many more.

**Deep Learning techniques, like CNN**, are used to train the machine vision of the robots and prepare it to handle this wide range of events. Engineers at RSIP Vision are experts in both machine vision and **deep learning**: we have already carried successful projects involving detecting and classifying object under a wide range of occurrences.

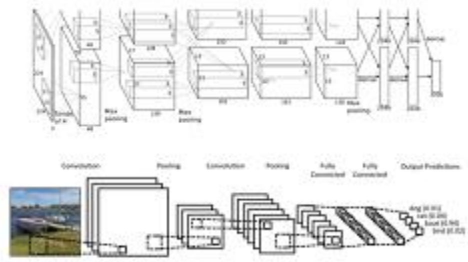
As we understand, robotics is not just the hardware and its mechanical parts: once you have the robot with its requirements, we can assist you in training your robot to do the task. We will use video or image feeds as a training set and we will program the required procedure, adapting the most proper deep learning scheme for the operation at hand.

Read more on our website about **Machine Vision for Robotics**.

Computer Vision News lists some of the great stories that we have just found somewhere else. We share them with you, adding a short comment. Enjoy!

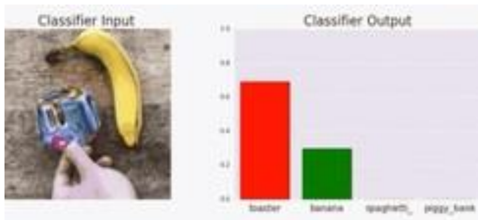
### 3D Face Reconstruction from a Single Image:

Let's start with something funny that you can try by yourself**!** This is a **3D reconstruction of the face** of @POTUS. How did we do it? We just used a lovely tool, courtesy of **Aaron Jackson**, PhD Student at the **University of Nottingham**. I tried to upload a pic of myself, but I think this one is way cooler. BTW, it takes only a few seconds and you can play with the result as you wish. **Try it here!**



### The Last 5 Years In Deep Learning:

Oh, no, not another story about **Deep Learning**! Well, this is a very interesting sum up of the hype, the impact, the progress and future perspectives of Deep Learning. We like this piece, courtesy of **Adit Deshpande**. **Read it here!**




### Stickers make CV software hallucinate things that aren't there

If you want to fool an AI vision system, this article is for you. **Adversarial images** (but also audio and text) distorts visual features to exploit weaknesses in the way computers look at the world to make them see stuff that isn't there. Kind of **optical illusions for Computer Vision Software**. **Read Now...**

### How can A.I. Generate Believable Fake Photos:

Sorry, guys: the lady in the photo looks familiar but she does not exist. She was generated by software under development at NVIDIA in no less than 18 days of image processing. This software can analyze thousands of (real) celebrity snapshots, recognize common patterns, and create new images that look much the same - but are still a little different. What will be the effect of this work on the spread of misinformation online, you can only guess. **Read Now...**

### How AI is transforming the criminal justice system:

This is far more serious. Should judicial courts use AI systems to speed, streamline or reduce bias in **criminal sentencing**? Believe it or not, they already do. But far from being a simple idea, it is actually very intricate, as it involves many severe difficulties: think only at how different the effects of two separate sets of training data would be! **Read Now...**

**Do you always hear that 'correlation does not imply causation'? Here is the brilliant proof!**

**Go package for computer vision using OpenCV 3.x and beyond: GoCV**

# Google Panorama Photo Fail

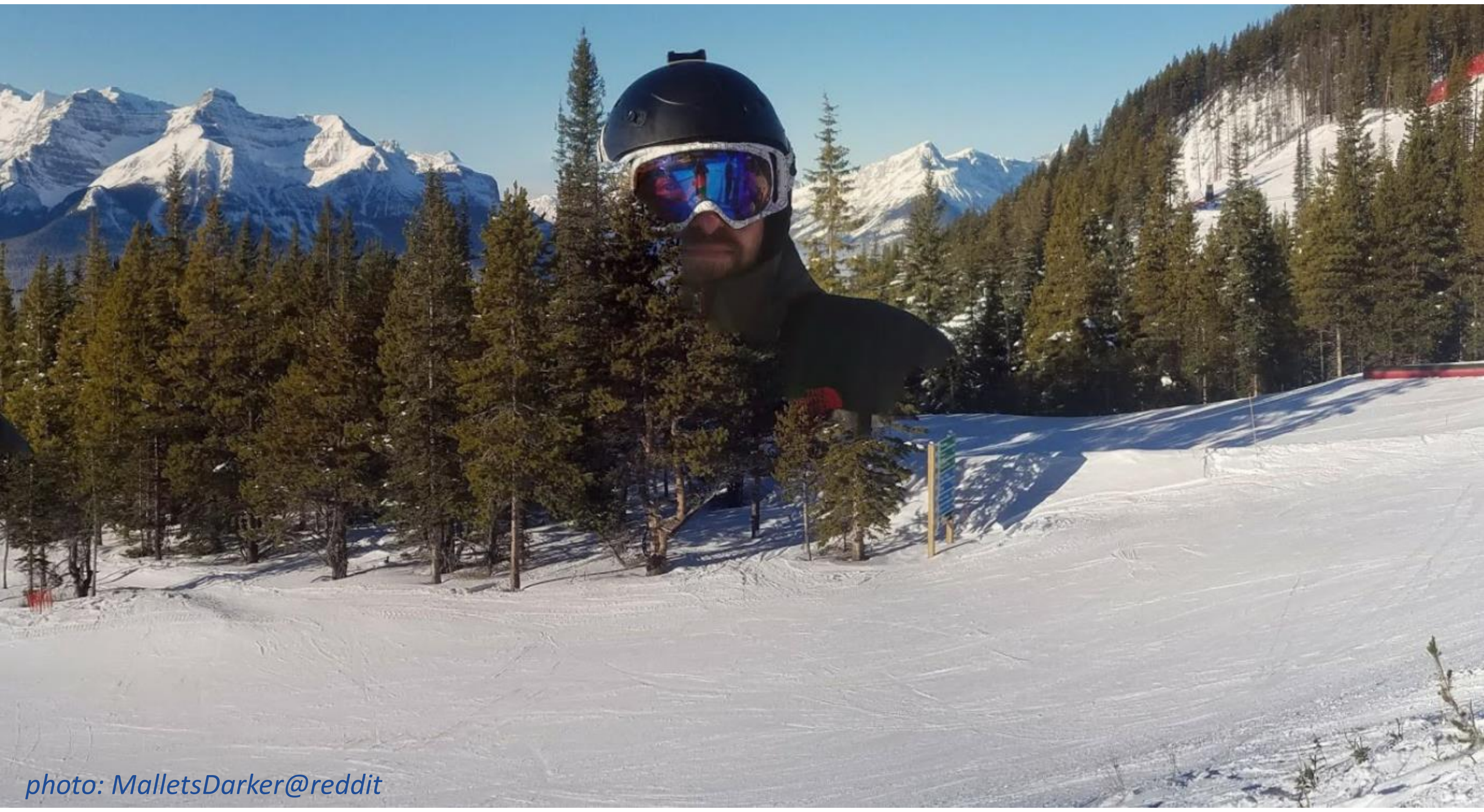


photo: MalletsDarker@reddit

It all started with a few friends taking pictures during a ski vacation at Lake Louise. When **Google photos** tried to stitch their 3 photos together, the result was hilarious. The photographer MalletsDarker@reddit decided to post it on Reddit and as you will see he was soon imitated by others.

This shows, as says very well Steve Dent on Engadget, "*how good AI can be at weirdly specific tasks and how bad it is at seeing, well, the big picture*". Like the insignificant fact that "*humans are not eighty feet tall*"...

You can read Steve's excellent comment about the reasons for Google's stitching fail.

What we can add here, is that this epic fail generated a whole set of posts on **Reddit**, each claiming that their fail is worst (see below). They remind me that when we launched **Computer Vision News** two years ago (!), we considered including also a "fail" section, open to our readers' posts. While we finally did not do it, it turns out that now Reddit has a whole subreddit for this. We only thought about it, Reddit did it…

by Assaf Spanier

*"TF-Slim wrappers reduce a very lengthy, complex code in TensorFlow to just a few lines"*

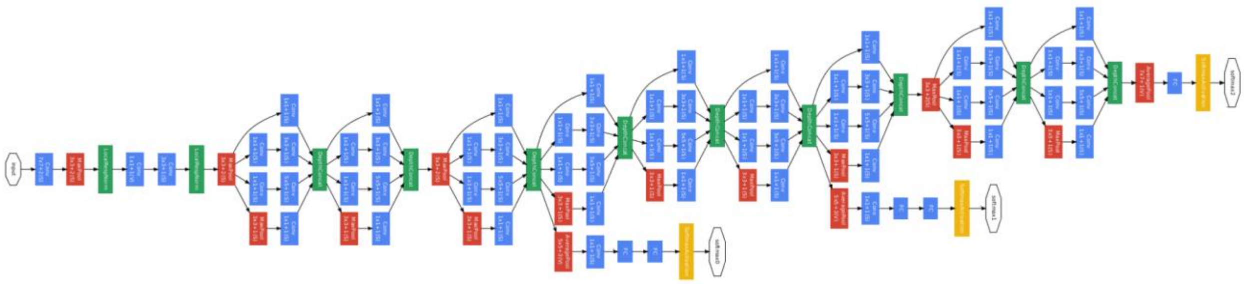
The last five years have seen an explosion in deep learning, which allowed huge advances in almost every application field. For us, one of its main benefits is its effect on various visual tasks. Together with these developments, the explosion in ready-to-use libraries for implementing deep learning might be confusing when contemplating which library will best suit your particular needs. This month we are having a look at **TensorFlow-Slim**, focusing on the **VGG** and **GoogLeNet** inception network implementation.

First, let's briefly recap **GoogLeNet** and **inception units**.

The idea of GoogLeNet is constructing a network from basic convolutional units. This should make the network computationally more efficient. The network is constructed by serially chaining together small, efficient units known as inception units, with only a single fully connected layer at the end.

Tool

Auxiliary classifier units are added along the chain, to allow viewing intermediate results for improved training of the network based on these results.



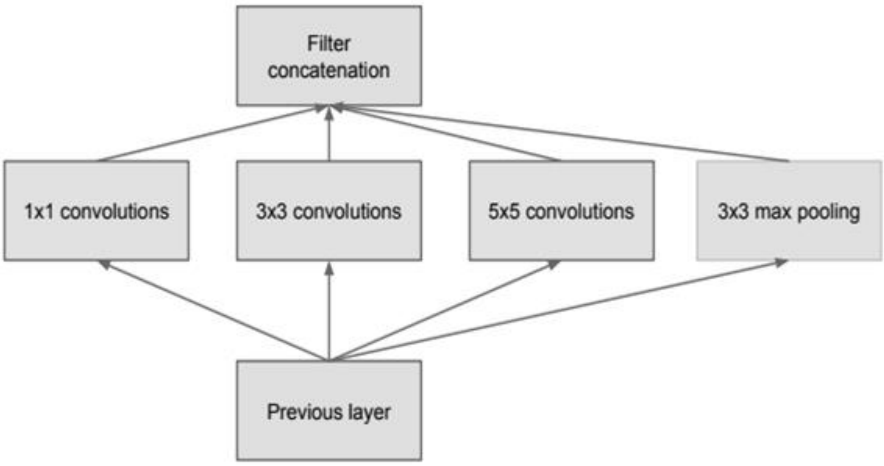
GoogLeNet has 12 times fewer parameters compared to **AlexNet**, despite being a much deeper network. Google has published 4 versions of GoogLeNet over the past two years, with the main difference between versions being variations in the inception units they use.

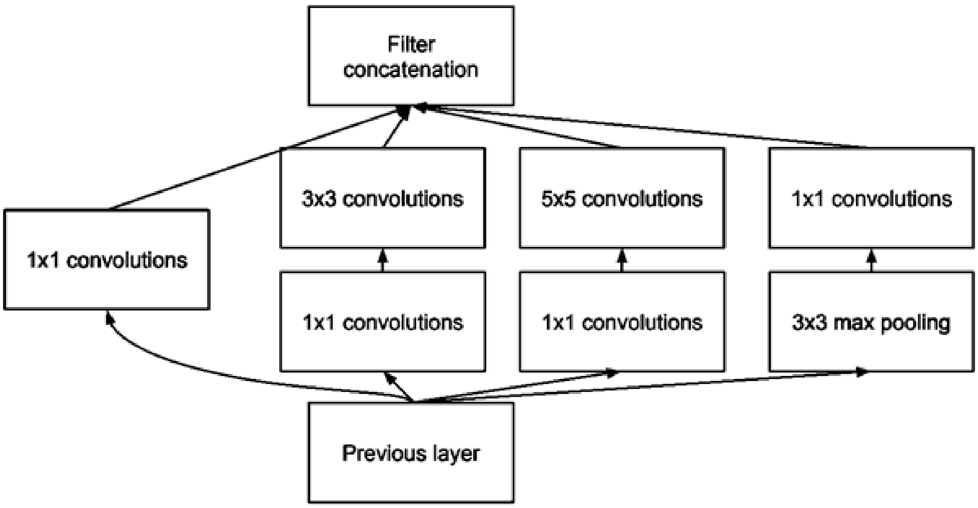
Let's look at the basic structure of inception units:

**Inception-v1:** The idea behind the inception unit is that it is a miniature network, so you can actually think of GoogLeNet as a network made up of a multitude of tiny networks.

The inception network runs its input in parallel through two types of operations: convolutions of different scales and max pooling -- then concatenates their outputs.

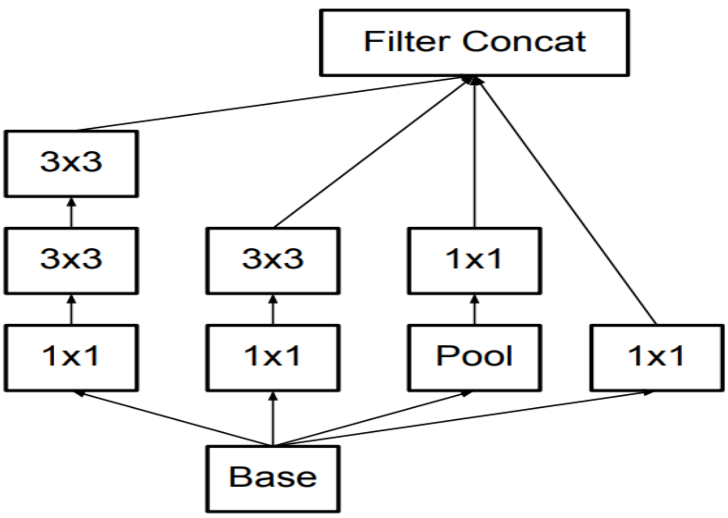
To illustrate how the inception unit works, we'll first look at a naïve implementation: in the figure at the right, the input is passed for parallel processing to three convolution units and a max pooling unit. The outputs of all 4 parallel units are concatenated together to form an output matrix of massive depth.



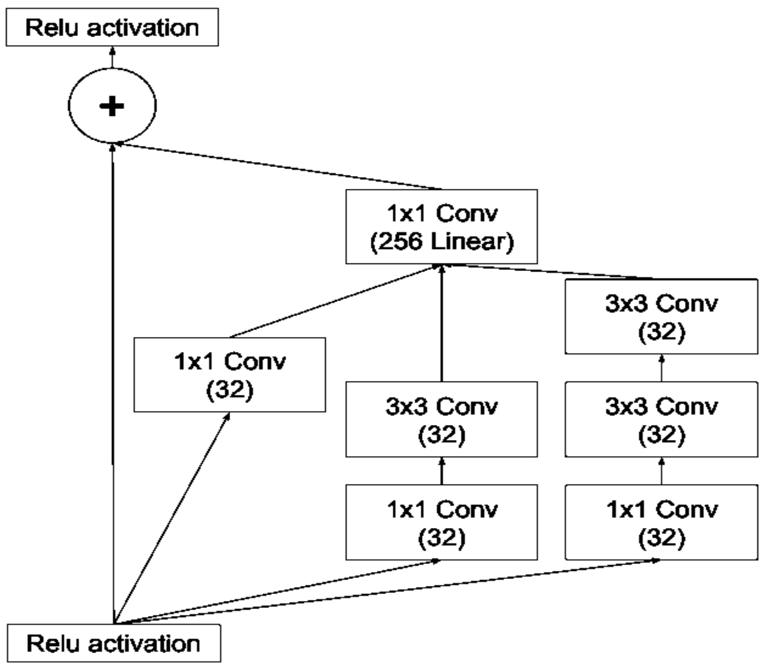

To reduce the volume of computations, the actual implementation includes conv units with 1x1 filters before each conv unit with non-1x1 filters and after the max pooling unit, cutting the number of operations in half.

**Inception-v2** is a variant of Inception-v1. It uses smaller conventional kernels, for example replacing the 5x5 conv unit by two chained 3x3 conv units, which reduces the number of parameters, memory use and number of operations.



**Inception-v3** is a variant of Inception-v2 with the auxiliary classifier units normalized per batch. A few other minor changes include using the RMSprop optimizer and label smoothing.

**Inception-v4** is a variant of Inception-v3. Inspired by the **ResNet** architecture, residual connection was added to the inception units. As you can see below, a residual connection was added alongside the existing Inception-v3 units, passing the input as-is to the unit after it.






## TF-Slim:

**TensorFlow** (see here and here) is an open-source software library for dataflow programming across a range of tasks. It is mainly used for machine learning with an emphasis on deep learning research and production. It provides good control of your network structure and functioning, with access to the inner workings of your deep neural network, such as the ability to directly update weights and gradients. Among the more advanced tools included in TensorFlow are highly flexible higher-level constructs, including Estimator, Experiment, and Dataset, which help you set-up your learning process in an easier way. Additional tools include: queues for computing tensors asynchronously in a graph, parallel-thread computation to speed up training. TensorFlow ships with powerful debugging tools, providing insight into internal structure and states.

TF-Slim is a super-lightweight library written using a **Functional Programming style**, that can be used right alongside any of TensorFlow's native operations. TF-Slim is a library that makes building, training and evaluating neural networks simple: the use of argument scoping and high level layers and regularizers allows the user to define models much more compactly. These tools increase both readability and maintainability; they also reduce the likelihood of error. Slim makes it easy to extend widely used computer vision models (e.g. VGG, AlexNet), either using them as a black box or use transfer learning techniques to warm start training algorithms.

To code a CNN layer in TensorFlow, you must define several low level operations:

1. Create the weight and bias variables
2. Convolve the weights with the input from the previous layer
3. Add the biases to the result
4. And apply an activation function.

This can be rather laborious to code:

```
input = ...
with tf.name_scope('conv1_1') as scope:
      kernel = tf.Variable(tf.truncated_normal([3, 3, 64, 128],
dtype=tf.float32,
                                               stddev=1e-1), name='weights')
       conv = tf.nn.conv2d(input, kernel, [1, 1, 1, 1], padding='SAME')
       biases = tf.Variable(tf.constant(0.0, shape=[128], dtype=tf.float32),
                            trainable=True, name='biases')
       bias = tf.nn.bias_add(conv, biases)
       conv1 = tf.nn.relu(bias, name=scope)
```

TF-Slim provides wrappers defined at the neural-network-layer level, streamlining setting up your neural network, while keeping all the options of TensorFlow available. For example, the following embodies the above TensorFlow code in TF-Slim:

```
input = ...
net = slim.conv2d(input, 128, [3, 3], scope='conv1_1')
```

TF-Slim also provides the useful function repeat, which you can call instead of defining your layers one by one or inside a loop, as you can see in this snippet:

```
net = slim.repeat(net, 3, slim.conv2d, 256, [3, 3], scope='conv3')
```

One final Slim shortcut we will look at is the `arg_scope` function, which allows the user to declare a set of variables to be used by the succeeding functions inside the `arg_scope` function, as demonstrated in the snippet below:

```
with slim.arg_scope([slim.conv2d], padding='SAME',
                    weights_regularizer=slim.l2_regularizer(0.0005)):
       net = slim.conv2d(inputs, 64, [11, 11],  scope='conv1')
       net = slim.conv2d(net,    128,[11, 11],    scope='conv2')
```

TF-Slim wrappers reduce a very lengthy, complex code in TensorFlow to just a few lines. For example, the entire VGG architecture is defined by the following snippet:

Tool

```
def vgg16(inputs):
  with slim.arg_scope([slim.conv2d, slim.fully_connected],
                      activation_fn=tf.nn.relu,
                      weights_regularizer=slim.l2_regularizer(0.0005)):
    net = slim.repeat(inputs, 2, slim.conv2d, 64, [3, 3], scope='conv1')
    net = slim.max_pool2d(net, [2, 2], scope='pool1')
    net = slim.repeat(net, 2, slim.conv2d, 128, [3, 3], scope='conv2')
    net = slim.max_pool2d(net, [2, 2], scope='pool2')
    net = slim.repeat(net, 3, slim.conv2d, 256, [3, 3], scope='conv3')
    net = slim.max_pool2d(net, [2, 2], scope='pool3')
    net = slim.repeat(net, 3, slim.conv2d, 512, [3, 3], scope='conv4')
    net = slim.max_pool2d(net, [2, 2], scope='pool4')
    net = slim.repeat(net, 3, slim.conv2d, 512, [3, 3], scope='conv5')
    net = slim.max_pool2d(net, [2, 2], scope='pool5')
    net = slim.fully_connected(net, 4096, scope='fc6')
    net = slim.dropout(net, 0.5, scope='dropout6')
    net = slim.fully_connected(net, 4096, scope='fc7')
    net = slim.dropout(net, 0.5, scope='dropout7')
    net = slim.fully_connected(net, 1000, activation_fn=None, scope='fc8')
  return net
```

**V4 in TF-SLIM:** To illustrate the efficiency of TF-Slim, let's look at the code of inception block 35 in the snippet below. See how short the definition of an entire deep complex network like GoogLeNet inception v4 becomes in TF-Slim, here.

Tool

```
def block35(net, scale=1.0, activation_fn=tf.nn.relu, scope=None, reuse=None):
  """Builds the 35x35 resnet block."""
  with tf.variable_scope(scope, 'Block35', [net], reuse=reuse):
    with tf.variable_scope('Branch_0'):
      tower_conv = slim.conv2d(net, 32, 1, scope='Conv2d_1x1')
    with tf.variable_scope('Branch_1'):
      tower_conv1_0 = slim.conv2d(net, 32, 1, scope='Conv2d_0a_1x1')
      tower_conv1_1 = slim.conv2d(tower_conv1_0, 32, 3, scope='Conv2d_0b_3x3')
    with tf.variable_scope('Branch_2'):
      tower_conv2_0 = slim.conv2d(net, 32, 1, scope='Conv2d_0a_1x1')
      tower_conv2_1 = slim.conv2d(tower_conv2_0,48,3, scope='Conv2d_0b_3x3')
      tower_conv2_2 = slim.conv2d(tower_conv2_1,64,3, scope='Conv2d_0c_3x3')
    mixed = tf.concat(axis=3, values=[tower_conv, tower_conv1_1, tower_conv2_2])
    up = slim.conv2d(mixed, net.get_shape()[3], 1, normalizer_fn=None,
                     activation_fn=None, scope='Conv2d_1x1')
    scaled_up = up * scale
    if activation_fn == tf.nn.relu6:
      # Use clip_by_value to simulate bandpass activation.
      scaled_up = tf.clip_by_value(scaled_up, -6.0, 6.0)

    net += scaled_up
    if activation_fn:
      net = activation_fn(net)
  return net
```

**Inception v4 in Keras:** If what you need is to quickly and efficiently build and test a neural network, with minimal coding required, go with Keras. In just a few minutes, you can construct simple or very complex neural networks. Keras is a very easy-to-use, modular front-end library, to TensorFlow's back-end, capable of running on top of other Machine- and Deep-Learning libraries like MXNet, Deeplearning4j, Tensorflow, CNTK or Theano. Keras library includes common neural network elements such as layers, objectives, activation functions, optimizers, etc. In addition it includes auxiliary functions for easily handling images and text.

Looking at the code snippet for inception block 35 below, we see it is slightly more elegant, and simpler to use, than even TF-Slim. However, the fact that with the mildly greater effort spent to code the network in TF-Slim, you get access to all of the powerful tools of TensorFlow should you need them, makes it worthwhile. For the entire GoogLeNet inception v4 network coded in Keras, see here.

```
def inception_resnet_block(x, scale, \
                           block_type, block_idx, activation='relu'):
    if block_type == 'block35':
        branch_0 = conv2d_bn(x, 32, 1)
        branch_1 = conv2d_bn(x, 32, 1)
        branch_1 = conv2d_bn(branch_1, 32, 3)
        branch_2 = conv2d_bn(x, 32, 1)
        branch_2 = conv2d_bn(branch_2, 48, 3)
        branch_2 = conv2d_bn(branch_2, 64, 3)
        branches = [branch_0, branch_1, branch_2]
    elif block_type == 'block17':
    elif block_type == 'block8':

    block_name = block_type + '_' + str(block_idx)
    channel_axis = 1 \
          if K.image_data_format() == 'channels_first' else 3
    mixed = Concatenate(axis=channel_axis, \
                        name=block_name + '_mixed')(branches)
    up = conv2d_bn(mixed,...)
    x = Lambda(lambda inputs, scale: inputs[0] + inputs[1] * scale,
               output_shape=K.int_shape(x)[1:],
               arguments={'scale': scale},
               name=block_name)([x, up])
    if activation is not None:
        x = Activation(activation, name=block_name + '_ac')(x)
    return x
```

Tool

**UnifyID is a service that provides authentication based on a person's movements. Rather than using passwords to gain access, UnifyID identifies a person with unique factors such as how they walk, sit or type.**

***"Insisting on deep learning models in every phase of our pipeline is not something that has paid rich dividends."***



**Vinay Prabhu** is **UnifyID**'s Principal Machine Learning Scientist and he explains that they harness statistical signal processing and machine learning to solve the issue of a person's identity. Their technology questions the institution of passwords used for authentication. Instead, they believe in the institution of **implicit authentication** in which simply one's entire being acts as a password.

UnifyID takes into account factors such as the way a person walks, the constellation of Bluetooth devices in use, and other biometrics. They do this in such a way that the whole experience is pretty seamless, and more importantly, passive.

The technology does not require any additive action on the part of the user. As the person walks, the technology uses a machine learning pipeline that harnesses state-of-the-art deep learning modules to perform **deep gait authentication** (which they term as **GaitNet**). It extracts routine cycles before comparing them with the normative model that the system has learned.

The pipeline itself entails harnessing on-device signal processing, spatio-temporal analysis and, of course, deep learning to build a statistical profile of the user's normative behavior and thus enable passive, seamless authentication.

They classify the sensors used by their system to collect this data in two disparate categories. The first category is **inertial measurement unit (IMU) sensors**, also known colloquially as motion sensors that typically include the triad of tri-axial accelerometers, gyroscopes and magnetometers on the phone. The system also accesses the **non-IMU sensors** which includes sensors like GPS and Bluetooth sensors that are available on the phone. In some cases, they also support bleeding edge sensors available on the phone such as hydration sensors. However, the system primarily focuses on the GPS, Bluetooth, and inertial measurement unit sensors.

Of course, this means that the user must give them access to all of this information. However, if a user does not want to share certain information such as GPS, the UnifyID system puts a weight of zero in regards to that sensor. This gives complete control to the user.

Based on feedback from their users, many seem hesitant to allow access to so many

*UnifyID demo video from TechCrunch Disrupt*

different sensors. Vinay and his team fully understand this apprehension. They passionately advocate this new philosophy of authentication while remaining extremely careful and honest about which aspects of sensor data is available to them.

Using an amalgamation of ideas from cryptography and deep learning, UnifyID's authentication pipeline operates in a differential privacy regime that obviates the possibility of them turning into a '**biometric honeypot**'. Vinay and his team want to make this the industry standard, knowing the importance of security regarding biometrics. They also insist that this approach is markedly different from the active biometrics field where theft of a person's biometric such as iris or thumbprint might render that victim exposed for the rest of his or her life.

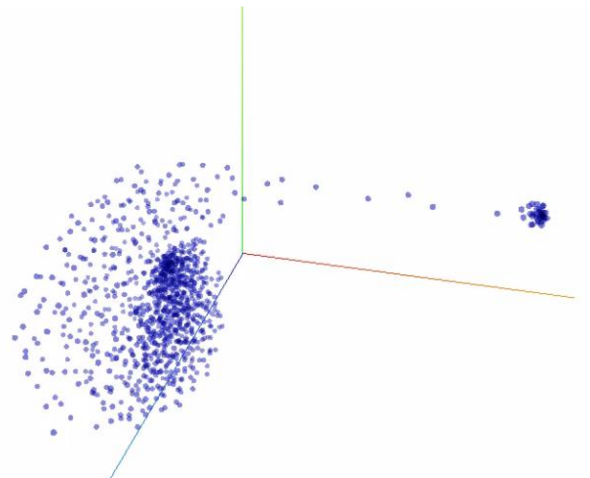
Vinay explains that the development of their application has two phases; the product side and the company side. On the product side, they have their SDK ready for integration with other apps, and many of their partner companies already have UnifyID embedded. Their system also provides GPS services.

They have developed their application using a **two-pronged strategy**. One is in terms of providing the confidence scores emanating from specific passive factors such as Gait, Bluetooth, etc. exposed via an API ensuring that users can handle the end product however they want. Each user may have their own security partners already with their own biometrics. They want to understand the level of confidence that they can get via gait or gaze, for example.

The other is a more holistic solution which they offer as an SDK for many of the activity apps. They will provide authentication as an additional service. Currently, they are in the final stage of collaborating with a large smartphone device manufacturing company as well as a big banking corporation providing a solution for ATM machines.

As far as the company is concerned, UnifyID just finished raising their series A from a top-tier venture capitalist firm in Silicon Valley. In terms of finance, they have made it through the feverish series A phase. They are good to go for the next few years in terms of money.

On the algorithmic side, Vinay details the four main components of their machine learning pipeline. **The first component**, termed broadly as the signal processing component, entails

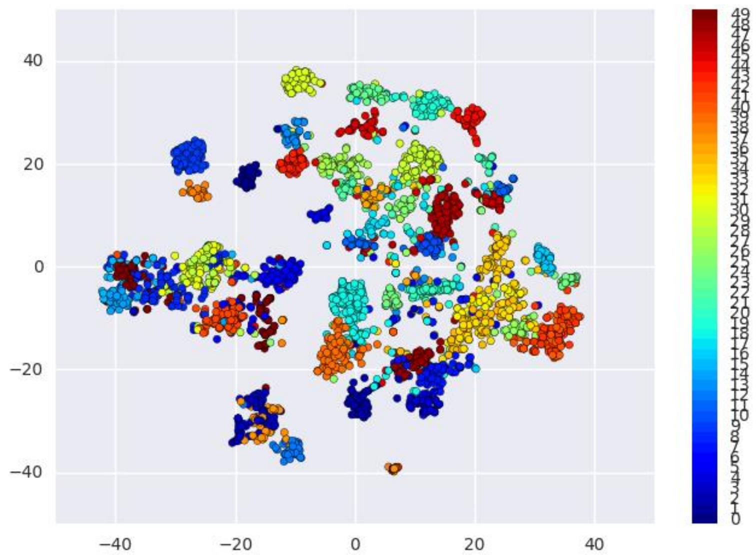

*Attacker rejection*

fast physics-inspired on-device pre-processing on noisy raw sensor data using state-of-the-art algorithms from the signal processing realm. This pre-processing component, he reveals, although currently under attack from those who espouse a strictly raw-data-driven end-to-end deep learning narrative, has been crucial in reducing the demand on the volume of training data required to train the forthcoming deep-learning component.

The second component, termed as the deep-feature-extraction component, exploits a novel **hybrid-CNN-LSTM-deep-learning** architecture that has a few 'convolution' layers that feed into specialized layers with LSTM cells in the last few layers. This component ingests the pre-processed sensor data from the previous signal processing component and churns out highly discriminative deep features that are then passed on to the forthcoming typicality modeling component in the pipeline.

The third component of their pipeline, the typicality modeling component, is in parts, inspired by the innovations such **Universal Background Models** and the i-vector approach that have attained much success in domains such as speech-based person identification and the general biometrics domain. The last component of their pipeline is the **decision-fusion-hypothesis-testing component** that entails both decision fusion across the several passive factors by using models from the menagerie of Graphical models as well as ideas from the Bayesian hypothesis testing body of literature to provide the final inference as well as the confidence scores associated with the inference(s). To sum up, they have three components: signal processing, deep learning, and hypothesis testing. The hypothesis testing ascertains the person's identity: *"At the end of the day* - Vinay comments - *we are trying to answer the question that, given all of the data, can it tell that the person behind the phone is indeed the person they are claiming to be?"*

In the process, they learned that implementing an end-to-end solution does not work. Different types of fancy LSTMs which are proposed didn't work either. The third thing is that the old, shallow algorithms like the Gaussian Graphical Models and Kernel machines are not to be thrown away. In fact, they play a very important role, especially when it comes to providing explainable inference. One of the things that they have learned through experimentations and implementations, going through several cycles, is not to be stubborn about models. In some cases, the most elegant model that can be put into production in the shortest available amount of time happens to be a 'shallow learning algorithm'. Vinay says: *"Embrace it. Don't chase deep learning, deep learning, deep learning*



**50-users gait clustering**

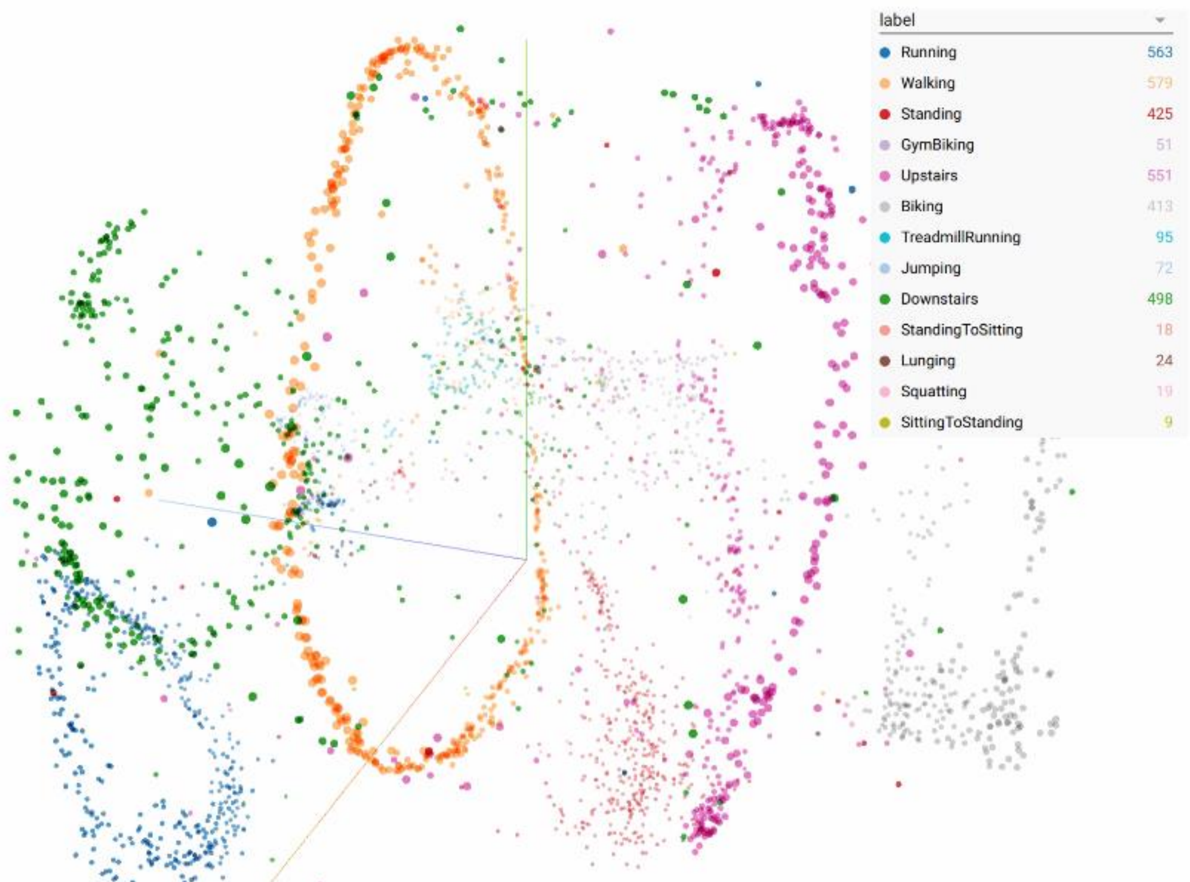
*as a catchphrase. Always take the most pragmatic position. Insisting on deep learning models in every phase of our pipeline is not something that has paid rich dividends. That is something that I would like to share with your readers."*

In case of false positives, when the system has not identified the right person properly, they have incorporated a cross-function into their pipeline: they developed the system taking into account the user experience, the **UI/UX**, especially for a person in desperate need of authentication. Today, UnifyID has a budget for UI/UX and they have conducted studies about it. They collected huge amounts of data from people of all different backgrounds. They studied how to set the hyper-parameters of their machine learning system in order to get an optimum point. Then they could send the hyper-parameters in such a way that **minimizes the false positive as much as possible**.

They also provide a hybrid mechanism, using **fallback authentication techniques**. They provide a seamless fallback on active authentication which can be as simple as asking a question. In some cases, their partners insisted on having the **option to revert to passwords**.

Through the development of their system, they learned that many people do not want to abandon the use of passwords. It's part of their routine. When they feel it is seamlessly authenticated, they become very nervous. They wonder how the technology knows enough to identify them. As a consequence, **UnifyID also allows for elegant fall back to active authentication techniques**.

Application




*Accelerometric data activity dimensionality reduction*

*The 3rd edition of Roy's book on Computer Vision receives attention at VIE (2005) from [Rita Cucchiara, a great friend of our magazine too]!*

**Roy Davies is Emeritus Professor of Machine Vision in the Department of Physics at Royal Holloway, University of London.**

**Professor Davies, you have observed the amazing growth in popularity of this field over the last few years. Since you entered into this field, how has it changed?**

I came into it around 1975, and more seriously by 1980. Back then, it wasn't even certain how to do the basics like edge detection and noise removal, which were at a very rudimentary stage. All sorts of people had all sorts of ad hoc ways of dealing with such operations. Often, Person A said, "*My method is better than yours!*", which was followed by "*No, it isn't!*", "*Yes, it is!*"… Clearly, this way of proceeding had no science in it. I was a physicist originally, and I needed proof. I did lots of experiments to try to ascertain the situation. Back then, I wasn't satisfied because I needed an underpinning theory to show how it works and how to optimize it. There's no use just having ad hoc solutions. You have to optimize the results. The only way to do that is to have a theory that will tell you what the parameters are. In fact, you have lots of parameters for each method, and you need theories that tell you how to optimize them.

**How could you measure the validity of a theory with the computer technology available in the seventies?**

Machine vision and computer vision started basically as image processing. For example, you look all over the picture for a speck of noise, which is like a black spot surrounded by a lighter area. What you were actually doing was applying convolutions and other simple operations, which didn't take much computation - so there was no bar to rigorous testing if one needed to do it.

**What major breakthrough over the years impressed you the most?**

The advance in deep neural networks has been absolutely phenomenal. Before 2012 I hadn't seen anything that had taken off at such a rate. I think the real reason for that is that a whole lot of preparatory work had already been done on artificial neural networks in the

*Ian Hannah, Darrell Greenhill, Roy Davies and Daniel Celano in Roy's lab (~1993)*

1990s. It was useful work, but the truth is that at that stage they weren't reliable. Also, people didn't know what these networks were 'thinking' or how they worked.

Gradually, people moved back to conventional algorithms instead. There were statistical pattern recognition methods that could emulate the neural networks. Meanwhile, after their demise towards the end of the 90s, people had continued working with them. It was a sort of underground movement, almost invisible to many people.

Suddenly, they emerged again in 2010 or so as 'deep' networks. By 2012-2015 they completely exceeded people's expectations for applications such as face recognition. They actually performed better than conventional algorithms, which was a shock. I, personally, like many other scientists, didn't like this because, again, you didn't really know how they were working. They seemed to be working, and working well, or even superlatively. And this time you couldn't ignore them. In those three or four years, the whole subject had changed radically. Now almost everyone wants to work with these new, deep neural networks.

**You are certainly aware of the article by [Nikos Paragios] about what he calls "[The Deep Depression]", in which he declares that - notwithstanding the impressive performance of deep learning - the progress of science is understanding what we are doing, rather than putting layers and adjusting layers. What is your take about that?**

Well, I read his article, and other people think exactly the same. I almost thought the same too. The thing is that now neural networks - the new sort, the deep ones - have actually taken off in such a way that you cannot ignore them. The question is: how can science develop in that atmosphere?

What we've got to recognise is that science advances in phases. You can go so far with one phase then you've got to take over with another. The thing I realized is that these neural networks were trained with millions of sample patterns, like faces from the internet. You need a million faces, or even a hundred million faces to train a network properly. But really, you must compare it with a human brain … the way children learn. They learn by playing with things and seeing everything going on all around them. They see faces, small and large, from every angle. They are learning from thousands of millions of faces in different positions. And they see the same face from different angles with thousands of millions of examples. It's difficult for deep neural networks to get that much data autonomously. Then what happens is that the human operator has to spend his time developing methods for providing the data – typically using tricks such as chopping images into several patches.

*Bob Fisher, Mark Nixon and Roy Davies exhibit their IAPR Fellowship awards (2008)*

*"Deep learning networks provide science with a valuable existence theorem which beckons the way forward"*

Nonetheless, the science of it is, to achieve these feats of recognition, you do need that much training on real data, with natural variation and real changes in lightness and darkness and contrast and so on. Neural networks have actually managed that in a fair number of cases. Face recognition is a very important case. What we've got to do is to try to get proper science to embody all of this. But it's almost impossible for a scientist to define what a face ought to look like, so the only way forward is to train scientific theories on similar quantities of real data, and to identify any theoretical short-cuts that validly advance the underlying methodology. Specifically, the data has to be modelled scientifically: this is now the 'missing link'. In summary, the value of deep learning has been to involve us in a phase of scientific discovery, but now a phase of deeper scientific understanding needs to follow.

**How do you compare the current generation of students with your generation? What do you think they could learn from the previous ones?**

Well, one of the things that has happened in recent years is that methods like MATLAB and Python have taken off. You have libraries of vision algorithms available: say edge detectors and Hough transforms … a whole range, a complete panoply of vision algorithms. So what the students know is that they have to take bits, put them together, and not think too hard about it. It's just like using Lego. You could say that's cheating. They're not doing the real thing. They don't really know what is in the blocks they are using. You need to know not only how to put the blocks together but also what is inside them to get the best out of them. Furthermore, you need to design some of the blocks in your own way in order to optimise their operation and adapt them to new tasks. In general it's become too easy to re-use old blocks: proper training of students needs to include invention, not just assembly!

*"I owe a huge debt to Michael Baker, my PhD Supervisor, who died in 2017"*

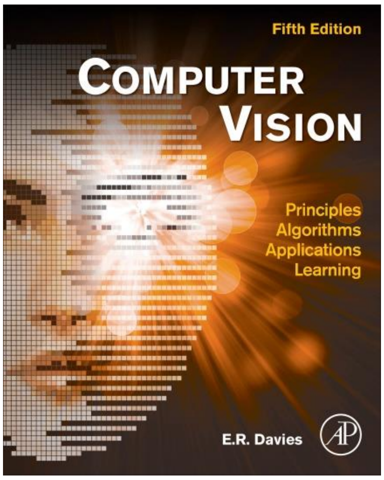
**Who was the teacher that impressed you the most?**

That's interesting, because my father was my teacher, role model and mentor until I was about 26. However, he wanted me to be a physicist, and I wanted to be an engineer, or even an inventor. At this point I started diverging from the path he had laid down. Also, as an undergraduate, my future PhD supervisor's lectures on electronics absolutely thrilled me. My hobby was electronics and still is for that matter. He filled in quite a lot of gaps that I didn't know. Anyhow, in the end, I chose his research lab because it had more electronics than any other lab I could find in Oxford's Clarendon Laboratory. There I worked at a great many frequencies between 0 Hz and 1010 Hz. All these electronics helped me

*Roy Davies, Frank Dellaert and Roberto Cipolla at BMVC (2013)* find a full-time position at Royal Holloway, where I still am.

However, Royal Holloway didn't have the liquid helium and other facilities that I had had at Oxford, so I had to rethink my career from scratch. Then, suddenly, I got lucky: my PhD student, Piers Plummer, managed to build a computer starting with a Motorola 6800 microprocessor kit costing just £70.

In fact, he not only built the computer but also produced a compiler so we could work on image processing using a tailor-made high-level picture processing language. This accelerated our work no end - at the same time completely changing my career!




**I know that you are now publishing a new edition of your book Computer Vision. Can you tell us about the new content in this fifth edition, compared to the previous ones?**

When you publish a new edition, you need to add about 25% of new material: basically, you throw away 25% of older, less relevant material, add 25% that is state of the art, and revamp the rest. That process went on for some time until the fourth edition. Then, after a mere 4 years my editor suggested producing a fifth edition. At first, that seemed a bit 'over the top', but I went along with the suggestion, and soon found that deep neural networks had 'taken off' immediately after the publication of the fourth edition. So I had the demanding job of bringing the Machine Learning and Deep Learning Network aspects bang up to date. Three new chapters were required to spell out the principles and methods and to put what people had achieved into proper perspective. It was an amazing experience and I also ended up with a completely new chapter on face detection and recognition highlighting the impact of deep learning. I ought to add that there was a related new chapter on object segmentation and shape models, which gave a practical demonstration of hand movements. Overall, I aimed to give a non-hyped view of the place of deep learning in the scheme of things, and to air the underlying scientific problems - and in particular to re-emphasise the vitally important place of probability in machine learning. All these changes necessitated a lot of remoulding of the whole text, so that its distinctive character was not lost and so that the reader could discern the increasing number of ways of segmenting images and locating a wide variety of real objects, from faces to road markings.

*Celebration dinner after Georgios Mastorakis' successful MPhil viva: left to right: Roy's wife Joan with Mina, Georgios and Roy (2009)*

# Workshop And Challenge On Learned Image Compression

Every month, Computer Vision News reviews a **challenge** related to our field. If you do not take part in challenges, but are interested to know the new methods proposed by the scientific community to solve them, this section is for you. This month we have chosen to review the **CLIC - Learned Image Compression** workshop and challenge, organized around CVPR 2018 which will be held later this year in Salt Lake City, Utah. The website of the challenge, with all its related resources, is here. The challenge is co-sponsored by Google, ETH Zurich and Twitter.



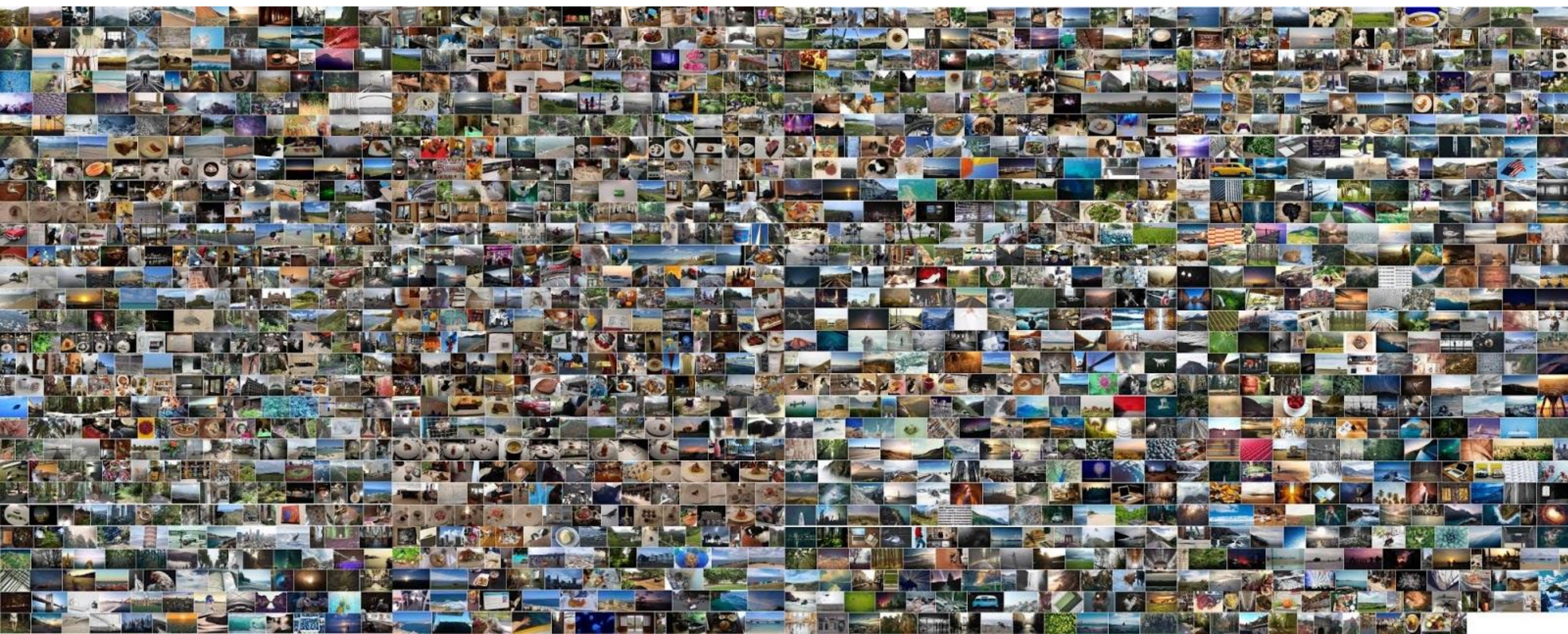
Image compression is a key procedure for any image today. Without it, each 12-megapixel image would require 36 megabytes of **storage space**, rendering it extremely tedious and resource-consuming to upload, download, exchange and store any digital photography.

**Compression techniques** have given satisfactory results during decades, with standards like **JPEG** and others.

The organizers of the challenge believe that breakthrough **neural networks** techniques can grant additional advancement to modern image compression standards. Hence, they are calling the machine learning community to join this effort. A **training set** of 1,633 uncompressed images (see them at the bottom of this page) has been recently released and the test set will be released on February 15.

Competing team are requested to submit the **compressed versions of the test set** by April 22 (extended deadline); rankings will be established based on both **PSNR** and **human rating** by experts. The overall winner will be decided by an ad hoc panel which will take into account also the **runtime performance**.

Provided dataset is actually double: set P ("professional") and set M ("mobile"). The two sets are meant to be representative for images commonly used in the wild. Participants can choose to **train neural networks or any other methods** on provided datasets and/or additional data, such as **ImageNet** and the **Open Images Dataset**.



# Deep Learning: Thriving for More than 90%

**RSIP Vision**'s CEO **Ron Soferman** has launched a series of lectures to provide a robust yet simple overview of how to ensure that computer vision projects respect goals, budget and deadlines. This month we learn about a special situation with **Deep Learning: Thriving for More than 90%**. It's another tip for **Project Management in Computer Vision**.

As an **R&D manager** or **team leader**, you can very soon find yourself in a situation where your team shows good results from the [POC stage](#) of a [Deep Learning project](#).

The results might be very encouraging, in the range of 85%-90% success rate (**Precision / Recall**, i.e. Positive Predictive Value / Sensitivity). Still, you might wonder about the ways to reach the next level. Here we propose several directions to consider:

1. **Different network architectures**.
2. **More data for training**: when this is applicable and it does not cause excessive delay to the work schedule.
3. **Add pre-processing**: sometimes it can emphasize the important features; rotate and resize to canonical form to enable smaller training set.
4. **Add post-processing**: removing false results.
5. **Sort out errors from the training sets**: false negatives and false positives by human annotation, whether you used a Mechanical Turk or any other method.
6. **Cont.**: in some [lucky] cases, network performance may already be better than the human labels, which will need to be corrected!
7. **Different resolution** might play a crucial role: we prefer low resolution for higher speed; but we might need the details for the classification task.
8. **Combining 2 or more schemes**, that can contribute - when taken together - to produce better results.
9. **Examine different training objects**: One can train on part of the object that is more prominent and less ambiguous than the whole object.
10. **Fine-tune hyper-parameters**: try different learning rates, mini-batch size and momentum.
11. **Combining different loss functions**: this is generally helpful.
12. **Verification of data augmentation results**: it must look authentic and represent the multitude of object appearance. If it fails, we train the net for our artifacts.
13. **Visually examine a sample** of specific failed and successful cases. You may find specific issues or situations that call for a solution which is easily understandable by a human observer.
14. **Use better input data**: for instance, we, at **[RSIP Vision](#)**, have seen that coronal planes work better than sagittal planes for disk segmentation in spinal scans.

# Michela Paganini

**Michela Paganini is a PhD student at Yale University in the Department of Physics and an affiliate at Lawrence Berkeley National Laboratory in Berkeley, California. Michela is also a member of the ATLAS Collaboration at CERN.**

**...And also a future member of FAIR...**

It's official. I signed my contract. I haven't made a public announcement or anything besides perhaps on Twitter, but yes I will be a postdoctoral researcher at Facebook AI Research next year after graduation.

**Twitter is sort of a public domain.**

I guess that's true!

**Can you tell us about your work?**

I am a physics PhD student. A lot of my work revolves around using machine learning to facilitate physics searches at The Large Hadron Collider at CERN, which is the European organization for nuclear research. I'm currently involved in all sorts of projects in the ATLAS Experiment: we are trying to test machine learning algorithms to either speed up or simplify methods to make them even more powerful than they are at the moment.

**Why do you find this field particularly interesting?**

I think there are a lot of open problems, a lot of low hanging fruit as well. It's a very exciting field.

Obviously machine learning has been a part of the way in which we work at CERN for decades now. Certainly, it's nothing new. Deep learning, itself, has only really made an impact for maybe the last five, at most ten, years, I would say. There's a lot that can be done with better algorithmic thinking and better algorithmic design. The questions that we are trying to answer are also extremely interesting to me personally. They are maybe more abstract in the sense that they don't necessarily affect people's lives tomorrow, but there are deeper questions about the universe, about what matter is really made of, what this universe we live in is made of... where we all come from in all of this.

I think both the daily job and the big picture are very exciting to me. There's really so much to be done, and there are so many open questions that you never spend a day without something to do. As a matter of fact, perhaps even the most frustrating thing to me is trying to decide what to work on because people always come to me with a lot of brilliant ideas and a lot of cool projects that one could get involved with. Unfortunately, there is not enough time to think carefully about all of them.

**Now we are curious, too! What did you learn about the universe that we don't know? What matter is really made of?**

Well, to be fair, before I even joined CERN, certainly the biggest discovery in recent years has been the Higgs boson. That was in 2012, and I was still an undergraduate student, not working in physics. I can't claim that as one of my victories! We've learned a lot more about how to distinguish certain signatures of particles in our detector. What we're trying to do as experimentalists is to try and go look for those signatures. The issue that we have to deal with is that we cannot observe those particles as they come to be at the center of our collision point; we have to measure their decay products through the detector. As you can imagine, a detector is something with a certain finite granularity. A lot of the information gets lost in a way as particles propagate through the detector. The real questions that we are trying to answer are how do we best extract information from what we can actually measure, which is finite in a way.

The machine learning is helping us answer how we can detect particles that we know very, very well such as bottom quarks or charm quarks. How do we distinguish the two once they interact with the detector? They look very similar! They have similar properties, even though they are different particles.

**How does your work involve Computer Vision?**

In ATLAS we use computer vision techniques to identify particles instead of cars, objects, or pedestrians. Imagine our detector as a large high-definitional camera that takes pictures of the collisions: our data can then be transformed into image format and analyzed using CV-inspired methods. This is a relatively new way of looking at LHC data which brings us closer to detector-level information; previously, instead, this information was simply summarized into a set of engineered features. In recent years we have seen an exponential growth in papers in particle physics using convolutional neural networks on image representations of our data, but the question of how to best represent the unique nature of our datasets (whether as images, sequences, graphs, etc.) is still very much an open research question, with lots of recent contributions presenting new and innovative strategies.

**You're Italian.**

Yes, I am Italian. I grew up in a city called Busto Arsizio, near Milan.

**In the ATLAS cavern, in front of one of the endcap wheels of the ATLAS detector. Michela is proudly wearing the jersey of Pro Patria, the football team of Busto Arsizio.**

**Then we grew up in the same region. When did you discover that you wanted to dedicate your life to science?**

I think very, very early on, actually. My mom is a nurse. She had scientific training as well. I grew up with her, and as I was an only child, I would get bored at home. What can you do? As a child, I was very curious and would pick up my mom's textbooks. They were actually chemistry textbooks: huge, old-school textbooks. I started reading, but understood absolutely nothing. I found that very fascinating. She was always very supportive of me getting a scientific education and so was my dad, of course. I discovered very early on that it was something I was interested in. I've always had a very rational mind. I had lots of questions, as every child does, and I thought that science could answer a lot of those questions. I found it very, very exciting. Then I enrolled in a scientific high school, in Italy we call it Liceo Scientifico. They provided me with all of the basis necessary to succeed in college. I knew right away what I wanted to do in college.

I went to university in the States where it's customary to be undeclared or undecided for some years of your schooling. In college, you can explore a lot of different subjects. It's a brilliant


Scuba diving in Bora Bora

system as well, but it's different than what I was used to, so I thought I had to make up my mind right away. I decided to study astrophysics actually rather than physics per se - or particle physics, specifically. I applied to some colleges in the United States and was lucky enough to get into Berkeley, which I ended up attending. One of the criteria that I used to make my final decision on which college to attend was the fact that Berkeley has an astrophysics undergraduate program. The astronomy department was separate from the physics department, but they're obviously very close and well-connected. It's been a long trajectory for me: I was lucky enough to have this in mind for a long time, but I think there are all sorts of different paths to get to the same point. A lot of people discover their passion later on. That's what happened with me with machine learning. I had no idea what it was until five years ago.

***"Growing up is a lot of learning how to deal with the unknown and realizing that the questions you are asking are actually the right questions to be asking if you want to push the boundaries of human knowledge"***

**You said that you asked many questions as a kid. What was the toughest question?**

I think it probably had something to do with some of the questions that we are still trying to answer today in particle physics. We can go back to the $10^{-10}$ seconds after the Big Bang, $10^{-15}$ seconds after the Big Bang, etc. We progressively know less and less about what happened, but we have a degree of knowledge up to a certain point. Then we don't anymore. And there's the biggest question of all -- I don't even know if it's a rational question to be asking: at a certain point, we want to know about what existed before the Big Bang. What was there before?

**What was there before?**

What did space and time expand into? We always try to rationalize these questions in terms of what we can perceive, what we can rationalize, and what our brain is used to. Sometimes it just doesn't even make sense to ask those questions exactly in those terms.

**Do you have any guess?**

[*laughs*] The physicist in me is probably going to cringe a little bit because I don't actually know what the right answer to that is. I think the point is that time, as we conceive it, started then. There's no way of asking what was before. I'm actually not a theorist, and I don't think deeply about these questions anymore. Maybe as a child I was a lot more curious than as an adult.

[*laughs*] Michela kind of gave up on these sorts of questions. I don't have an answer for those, but I sort of learned how to live with that. It's still worthwhile to ask these sorts of questions, but it's ok to realize that, at a certain point, you get to the real edge of human knowledge. Growing up is a lot of learning how to deal with the unknown and realizing that the questions you are asking are actually the right questions to be asking if you want to push the boundaries of human knowledge.

**What function does curiosity have in your**

**day to day scientific motivation?**

I think it's really important, although my daily work revolves around coming up with ideas that, in my mind, are very simple. Then you have to teach your computer how to do that. That can take a lot longer. Once you do get results, that's when curiosity really starts playing a role again. Now you have to interpret results. We've come to a point where we have machines do all of the calculations, all of the tedious work, work that we don't want to do. Then out comes a plot. Now the physicists need to sit back and start questioning the plot and the results. Does it make sense? What is it telling me about the world? Is it in agreement with my hypothesis or is it not?

That's usually not straightforward. In general, you need to start thinking about what other plots you can make and what other studies you can make to figure out what came out of your previous experiment. Every once in a while, there is maybe a more pure form of curiosity, which is when you start wrapping up a project and jump into the next one. From there, you really start wondering: what you can do that has never been done before, what is the topic that people have wanted to tackle without necessarily knowing how to? That's when you can really think about the methods out there. How can I rotate them to fit my problem? That can be something very innovative.

*"I feel culturally European, rather than necessarily Italian"*

**Where is home, Michela? In Italy? In the States? Or is there no home yet?**

It's everywhere. I'm blessed with the opportunity of having lived in a lot of different countries throughout my life, primarily in Italy for the first 17 years of my life… then back and forth between the United States and Switzerland… France for a bit and, for a short while, in England. I feel culturally European rather than necessarily Italian. I feel at home in the United States as well. Now I live in San Francisco. I've spent almost a decade of my life in the United States, perhaps my most formative years. There is no one place that I call home. There are multiple places that have been important in my life.

**What do you enjoy besides science?**

I like to play tennis. I've been playing tennis ever since I was little, maybe when I was five years old or even less. Whenever I get the chance I go out and play tennis, squash, or any sort of racket sports. I also really enjoy watching tennis on TV.

**Then we have another thing in common, besides being Italian. Who**

With Yale ATLAS colleagues from the Tipton group

**is your favorite player?**

I like Roger Federer.

**That's too easy! I go for Simona Halep. Was it ever difficult for you to be a female scientist?**

I didn't feel difficulties in becoming a female scientist until later in my career, perhaps in college or graduate school. Italy has quite a different perspective on science having a very long tradition of excellence in science. I think a national pride for science maybe trumps the prejudice of it being a male dominated field. As I was growing up, I had a lot of support from my family, teachers, and classmates. I went to a scientific high school. We were about 50/50 in terms of boys and girls in my class. It never even occurred to me that science was not something for women.

My major inspiration was my mother. As it got more serious with my college and university career, I started noticing a deficit in the number of women. There were certainly instances that I can remember that were not easy, especially when it came to more practical skills that I did not necessarily have. I remember that my very first electronics lab in undergraduate was tragic. It was probably one of my worst experiences. Not that the class was bad. But I felt extremely out of place. I never really tinkered around with any electronic kits as a kid, or anything manual. I always thought of science as something very theoretical. I really felt out of place among a lot of other students that had had experiences with robotics in their American high schools. I, instead, was in a sorority and I definitely did not look like I belonged. It started becoming apparent to me. I'm sure it was apparent to many others at first. It was never a weight for me that I had to carry around. I think around that time after the second year of college I started to realize that I really don't look like I belonged in that department, but I did because I was a good student. In the end, it worked out well. It worked out fine, but it was hard.



At home in Milan, Italy

**Do you still have this feeling of not belonging?**

It's always there in a way. When you go to conferences, and 90% of the people are of a different gender. A majority of the authors you admire don't look anything like you or don't have the same issues to deal with on a daily basis. I think that is extremely problematic. I think it's something that is always in the back of everyone's minds. The only way to succeed is to really start believing in yourself. It's true that a lot of people don't look like you, but you do belong. You are smart: you have everything that it takes. It's an extra mental effort that you have to make every single day to remind yourself not to look around and just focus on what you are doing. Unfortunately, a lot of people have the privilege of not having to do that every single day, while others do.

**If you could fulfil any dream, what would it be?**

I wish there was a way for my family and my friends to be together again. It's obviously very complicated because everyone has their dreams and aspirations that take them all across the globe.

***"...spend more time learning what it is to be a good mentor!"***

**Is there one thing that you would like to change, to make our community better?**

STEM is very competitive in a nonproductive, toxic way. People try to get an edge over others in all sorts of ways using all sorts of methods. I'm not that sort of person. I would never do that. I can't even see myself doing something like that. Sometimes I'm left wondering if I have what it takes to succeed because I can't lie about my abilities. If I could change something, it would be the level of honesty of people in this field. Another aspect of our field that I would like to change is the mentoring system. I think we have truly excellent examples of mentorship in some subfields and subcommunities. For example, Women in Machine Learning have provided me with a lot of mentorship, connections, networking and encouragement! That was absolutely crucial for me to get to where I am. A lot of minorities are creating these sorts of groups. In the field, at large, there is really a lack of incentives for advisors to also be good mentors. I think to be a good advisor, you need to go far beyond checking whether a result is correct or providing a generic research direction. A lot of what it means to me to be a good advisor should be being able to provide connections in an academic world that is so cutthroat.

Thankfully, my advisor, himself, has been great. I was able to network with individuals that were real career sponsors. I am absolutely thankful to them. I realize that doesn't come as a standard for all students. This sort of system helps well-connected people advance, but doesn't necessarily let the true talents stand out. If I could change something, it would be that: to have people spend more time learning what it is to be a good mentor!

**Here and on the cover of this magazine: Michela is underground with the CMS detector.**


**AAAI conferences on Artificial Intelligence**
New Orleans, LA — Feb 2-7 — Website and Registration

**Robotic Vision Summer School - Australian Centre for Robotic Vision**
Kioloa, Australia — Feb. 4-9 — Website and Registration

**RE•WORK Intro to Machine Learning in Healthcare Workshop**
London, UK — Feb 14 — Website and Registration

**CardioFunXion Winter School 2018**
Lyon, France — Feb 19-22 — Website and Registration

**IEEE Conference on Integrating Vision and Language Processing**
Tartu, Estonia — Mar 5-7 — Website and Registration

**WACV Winter Conference on Applications of Computer Vision**
Lake Tahoe, NV — Mar 12-15 — Website and Registration

**Sustainable Smart Farming / Precision Agriculture Seminar**
Truro, Canada — Mar 15-16 — Website and Registration

**ACM/SIGAPP Symposium On Applied Computing**
Pau, France — Apr 9-13 — Website and Registration

**CLAP 2018: 1st Latin American Congress of Precision Agriculture**
Richardson, TX — Apr 11-13 — Website and Registration

**IWISC - Intl. Workshop on Interactive and Spatial Computing**
Richardson, TX — Apr 12-13 — Website and Registration

**AI Expo Global - Delivering AI for a Smarter Future**
London, UK — Apr 18-19 — Website and Registration

Did we forget an event?
Tell us: editor@ComputerVision.News

## FREE SUBSCRIPTION

Dear reader,

Do you enjoy reading Computer Vision News? Would you like to receive it **for free in your mailbox** every month?

**Subscription Form**
(click here, it's free)

You will fill the Subscription Form in **less than 1 minute**. Join many others computer vision professionals and receive all issues of Computer Vision News as soon as we publish them. You can also read Computer Vision News in PDF version and find in our archive new and old issues as well.


**We hate SPAM and promise to keep your email address safe, always.**

## FEEDBACK

Dear reader,

How do you like Computer Vision News? Did you enjoy reading it? Give us feedback here:

**Give us feedback, please** (click here)

It will take you only 2 minutes to fill and it will help us give the computer vision community the great magazine it deserves!