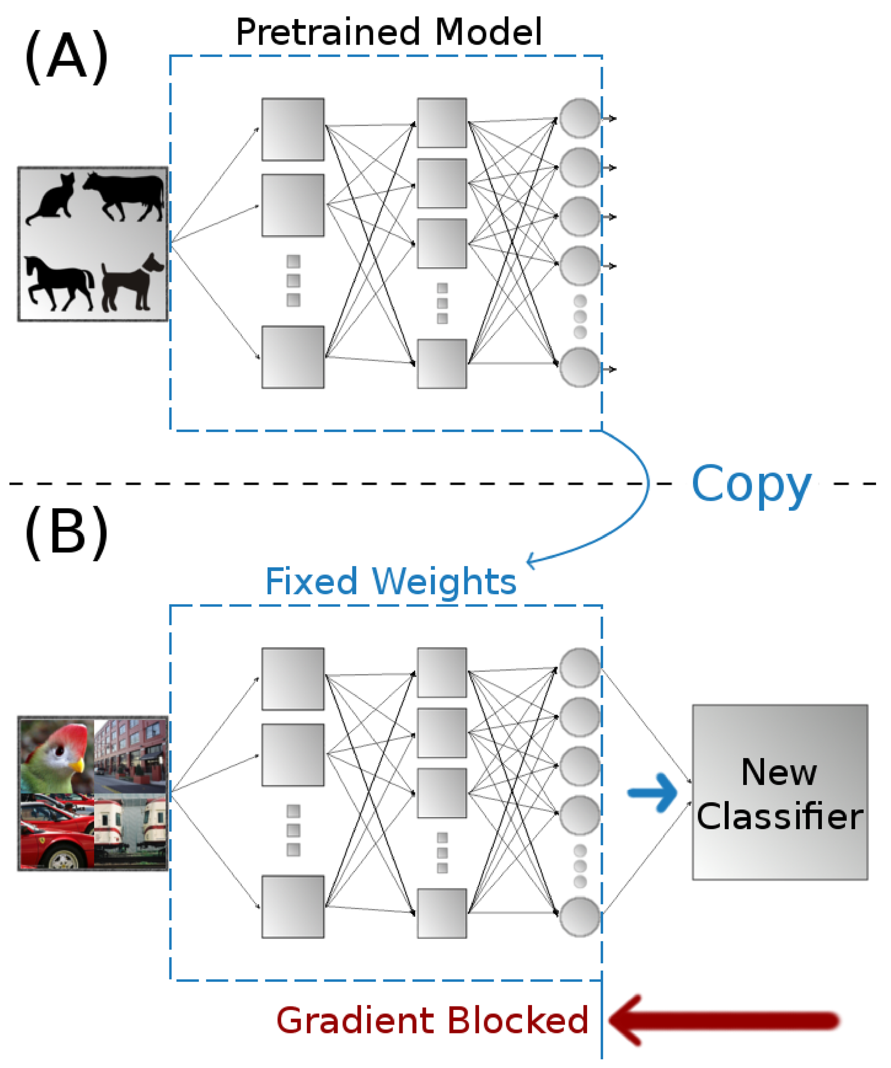
Deep Neural Networks Under Stress

Micael Carvalho[1,2], Matthieu Cord[1], Sandra Avila[2], Nicolas Thome[1], Eduardo Valle[2]
(1) Université Pierre et Marie Curie, UPMC–Sorbonne Universités, LIP6, Paris, France
(2) University of Campinas, UNICAMP, RECOD Lab, Campinas, Brazil
arxiv.org/abs/1605.03498

Context

Deep CNN's outstanding performance on visual tasks. Transfer takes advantage of it



A pretrained classifier **(A)** can be seen as a feature extractor, used by **(B)** for classifying a different task.

Questions
- Adequate representation space?
- Possible to compress?

Experimental Protocol

Main apparatus
- VGG-M from [1];
- Pascal VOC 2007;
- Linear SVM.

Supplementary experiments
- BossaNova from [2];
- GoogLeNet from [3];
- (paper-only) MIT-67 Indoor;
- (paper-only) UPMC Food-101.

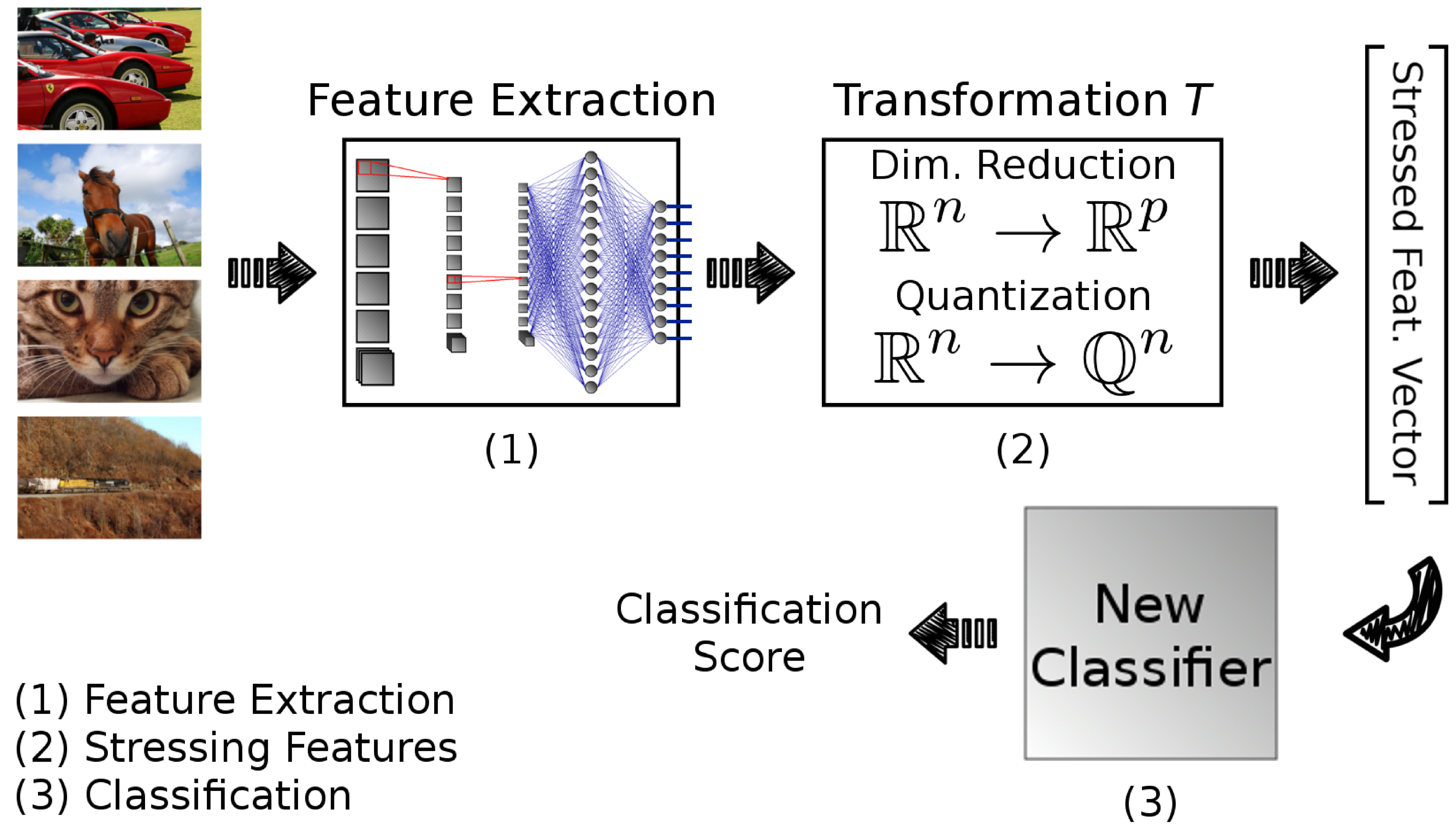
Original Scores

	VGG-M	GoogLeNet	BossaNova
VOC2007	76.95%	80.58%	51.02%
Dims.	4×10^3	5×10^4	6×10^4

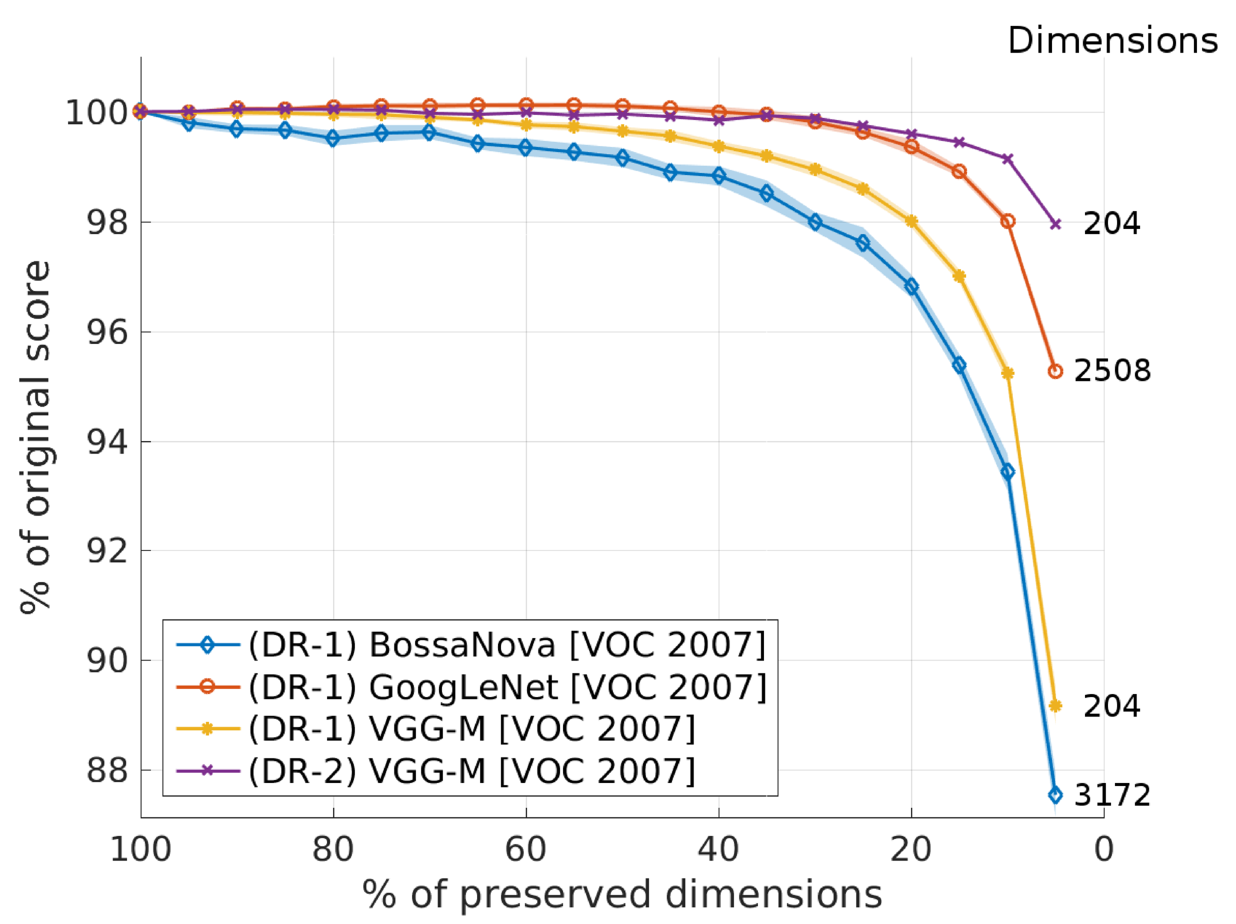
References

[1] K. Chatfield et al. *Return of the Devil in the Details: Delving Deep into (...)*, BMVC, 2014;
[2] S. Avila et al. *Pooling in Image Representation: The visual codeword point of view*, CVIU, 2013;
[3] C. Szegedy et al., *Going deeper with convolutions*, CVPR, 2015.

Overview



Dimensionality Reduction



Robustness
- VGG > BoVW. Its feature vector is 15-times smaller;
- GoogLeNet > VGG, but has 12-times more dims;
- **(DR-2)** > **(DR-1)**, as it preserves important info.

Removing 95% of dims.
- **(DR-1)** BossaNova 88%;
- **(DR-1)** VGG-M 89%;
- **(DR-1)** GoogLeNet 95%;
- **(DR-2)** VGG-M 98%.

w.r.t. the original scores.

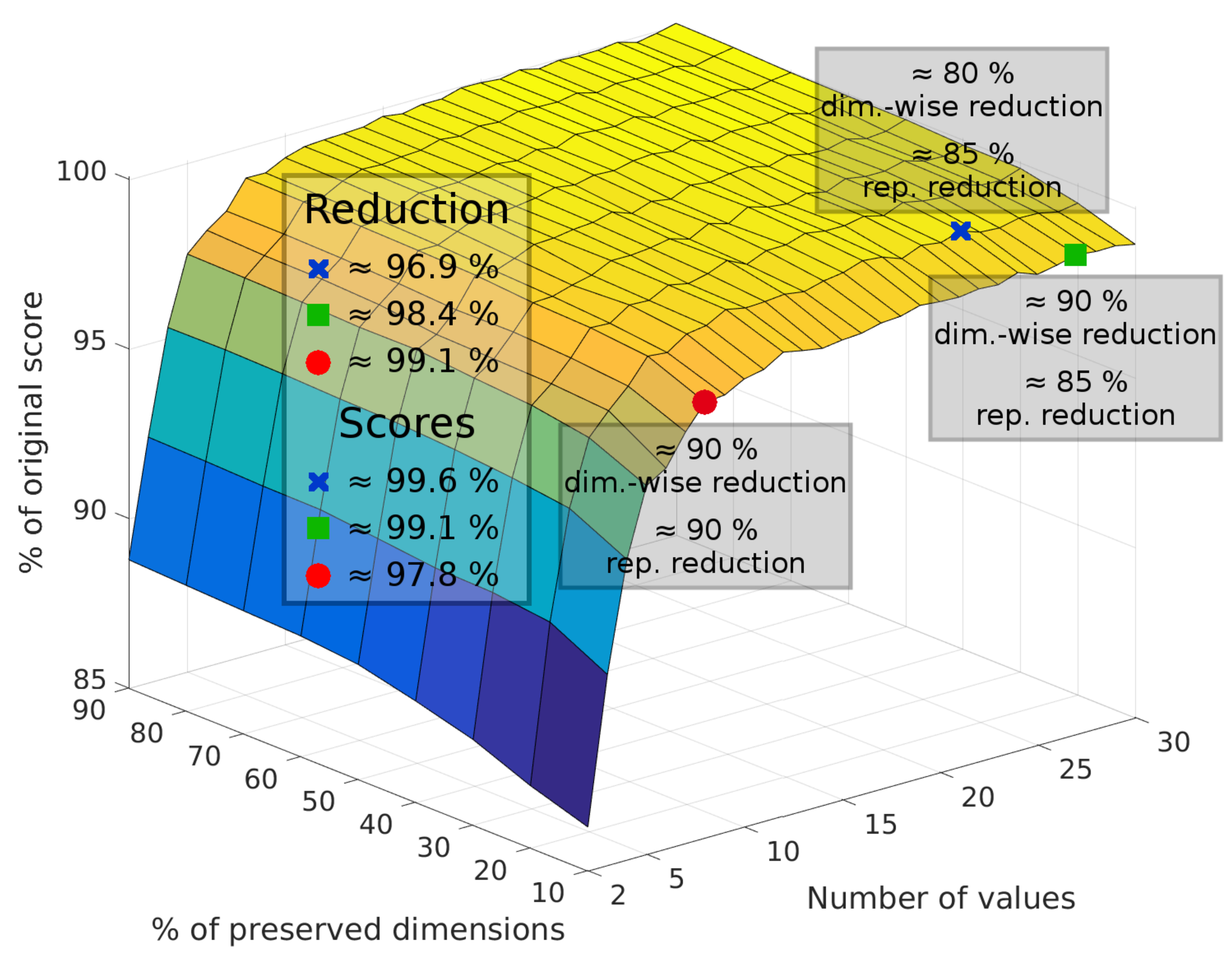
Feature Compression

Observations
- **(DR-2)** and **(Q-2)** have complementary properties;
- Their combination yields compact feature vectors;
- Useful for remote classif., image retrieval and more;

Detailed combination

(DR-2)	(Q-2)	Compr.	Score*
80%	85%	96.9%	99.6%
90%	85%	98.4%	99.1%
90%	90%	99.1%	98.7%

* w.r.t. the original scores.



Stress Framework

Quantization: In $h \in [1, 30]$ regular intervals, using the minimum (min) and maximum (max) scalar values in training set:

Q-1

$$st = \frac{max - min}{h}$$

$$\mathcal{H} = \{(min + \tfrac{st}{2}) + st \times i \mid \ 0 \leq i < h\}$$

$$T_{Q\text{-}1}(\mathbf{x}_{ij}) = \arg\min_y \{|\mathbf{x}_{ij} - y| \mid y \in \mathcal{H}\}$$

Q-2

$$st_t = \frac{max(\mathbf{x}_t) - min(\mathbf{x}_t)}{h}$$

$$\mathcal{H}_t = \{(min(\mathbf{x}_t) + \tfrac{st_t}{2}) + st_t \times i \mid \ 0 \leq i < h\}$$

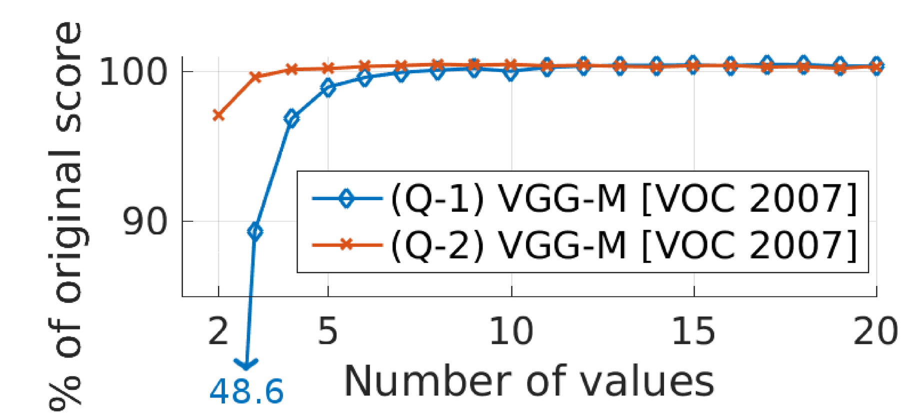
$$T_{Q\text{-}2}(\mathbf{x}_{ij}) = \arg\min_y \{|\mathbf{x}_{ij} - y| \mid y \in \mathcal{H}_j\}$$

Dimensionality Reduction: With n initial dimensions, at each step $1 \leq i \leq 20$ we preserve only

$$p_i = \lfloor \frac{n \times (21 - i)}{20} \rfloor$$

Dropping strategy: **(DR-1)** randomly and **(DR-2)** PCA-based.

Quantization



To keep original scores
- 7 values for **(Q-1)**
  $\lceil \log_2 7 \rceil = 3$ bits;
- 4 values for **(Q-2)**
  $\lceil \log_2 4 \rceil = 2$ bits.

Representing
- 9.4% of # bits for **(Q-1)**;
- 6.3% of # bits for **(Q-2)**.

Conclusion

Highlights
- Deep features are highly redundant;
- BoVW is not as robust as deep feat;
- Perf. depends on dataset complexity.

Source code available at
github.com/MicaelCarvalho/DNNsUnderStress