How far can you get with a modern face recognition test set using only simple features?

How far can you get with a modern face recognition test set using only simple features?

Nicolas Pinto
MIT
Cambridge, MA
pinto@mit.edu

James J. DiCarlo
MIT
Cambridge, MA
dicarlo@mit.edu

David D. Cox
The Rowland Insitute at Harvard
Cambridge, MA
cox@rowland.harvard.edu

Abstract

In recent years, large databases of natural images have become increasingly popular in the evaluation of face and object recognition algorithms. However, Pinto et al. previously illustrated an inherent danger in using such sets, showing that an extremely basic recognition system, built on a trivial feature set, was able to take advantage of low-level regularities in popular object [10] and face [11] recognition sets, performing on par with many state-of-the-art systems. Recently, several groups have raised the performance "bar" for these sets, using more advanced classification tools. However, it is difficult to know whether these improvements are due to progress towards solving the core computational problem, or are due to further improvements in the exploitation of low-level regularities. Here, we show that even modest optimization of the simple model introduced by Pinto et al. using modern multiple kernel learning (MKL) techniques once again yields "state-of-the-art" performance levels on a standard face recognition set ("Labeled Faces in the Wild" [7]). However, at the same time, even with the inclusion of MKL techniques, systems based on these simple features still fail on a synthetic face recognition test that includes more "realistic" view variation by design. These results underscore the importance of building test sets focussed on capturing the central computational challenges of real-world face recognition.

1. Introduction

The development of a robust face recognition algorithm capable of functioning in unconstrained, real-world environments will have far-reaching applications in our modern digital world. While considerable progress has been made towards building an artificial system that can match human performance, no clear solution has emerged. At the core of this challenge is the extreme diversity in viewpoint, lighting, clutter, occlusion, etc. present in real-world images of faces, which allows any given face to produce a virtually infinite number of different images. A successful recognition system will have to accurately recognize many individuals while tolerating these variations.

To guide any serious effort towards solving face recognition, one needs to define detailed specifications of what the problem is and what would constitute a solution, so that incremental progress can be precisely quantified and different approaches can be compared through a standard procedure. For the purposes of a recognition system, defining a specification amounts to choosing a test set against which an algorithm's performance is evaluated. Recently, it has become increasingly popular to evaluate models on large test sets of "natural" images [4, 5, 7]. Such an approach is appealing, as it is relatively easy to collect many images from the Internet, and it is relatively efficient to label them (e.g. [14, 20, 3]). However, there are significant downsides to this approach as well. Importantly, there is no guarantee that such a set accurately captures the range of variation (e.g. view, lighting, etc.) found in the real-world. A variety of factors conspire to limit the range of variation found in such image sets — e.g. posing and "framing" of photographs from the web, implicit or explicit selection criteria in choosing images for the set, etc. Images collected in this manner may also have subtle low-level confounds that "give away" the task, such as image artifacts or backgrounds that covary with face identity.

As a consequence, it is difficult to know if a given model achieves its recognition performance by robustly solving the problem (i.e., genuinely tolerating image variation), or by exploiting accidental low-level regularities present in the test set. This danger was recently demonstrated by the studies of Shamir [15] and Pinto et al. [11, 10] on popular face and object recognition test sets. Specifically, Shamir showed that relatively high performance was possible on various face recognition sets using image patches taken from the background, indicating that there was significant, diagnostic covariation of background content with face identity. At the same time, Pinto et al. demonstrated

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

that an extremely rudimentary algorithm was able to match or exceed the performance of many state-of-the-art vision systems (on the Caltech101 [4], Caltech256 [5], AR [22], ORL [26], CVL [23], YALE [28], and LFW [7, 24] sets). Interestingly, the same "null" model was easily defeated by ostensibly "simpler" synthetic recognition tests specifically designed to better span the range of real world variation. These results indicate that performance reports might better be judged relative to simple baseline models (e.g. based on pixels or wavelets) that are able take these low-level regularities into account.

Recently, with the advent of large scale machine learning techniques [18], it has become possible to significantly outperform the "trivial" baselines set forth in [11, 10] on several object and face recognition test sets. These approaches work by optimally combining many image features (e.g. [19, 6, 21, 2]). However, it unclear whether these approaches tap into some deeper solution to the underlying problem, or derive their increased performance from enhanced exploitation of low-level regularities.

To offer insight into this problem, we here apply a similar large-scale approach ("out-of-the-box" multiple kernel learning, [31]) to the trivial representations described in [10, 11]. Thus while the underlying representation ("front-end") remains unsophisticated in its processing of shape, lacking any mechanism to help tolerate image variation, we have added highly sophisticated "back-end" processing. We combine variants of the trivial features proposed by Pinto et al [11, 10] to investigate whether more low-level regularities can be captured using a large-scale (but not necessarily smarter) classifier backend. We evaluate this method on "Labeled Faces in the Wild", a large natural face recognition set publicly-available [7] and contrast the results with a small synthetic face recognition set, specifically designed to include controlled image variations [11].

2. Combining Trivial Features

In the following experiments, the processing of images was divided into two phases: a *representation* phase, in which images were transformed into feature vectors, and a *classification* phase. Since multiple kernel learning techniques (see below) rely on blending of multiple representations, we generated a series of variants based on two basic classes of representation:

1. *Pixels*: a representation based on raw pixel values (with optional spatial resampling, and Gaussian blurring)

2. *V1-like*: a simple representation inspired by the known properties of cortical area V1 [11].

2.1. Trivial Representations

2.1.1 Pixel-based Representations

Here, the *Pixels* representation is simply based on unrolling a preprocessed image into a n-dimensional feature vector. Simple preprocessing steps were added as follows:

1. use color information if present or convert the image to grayscale (2 variants: grayscale or color),

2. normalize the original image to have zero-mean and unit-variance,

3. blur the image with a Gaussian filter (3 variants: no blur, $\sigma = 1$, $\sigma = 2$).

By exhaustively crossing all possible variants of these three steps, one can produce up to six pixel-based feature representations (2 color spaces by 3 blurs).

2.1.2 V1-like Features

V1-like models are composed of a population of locally-normalized, thresholded Gabor wavelets spanning a range of orientations and spatial frequencies. For our purposes, these models are intended as "null" models, as they only represent first-order descriptions of the primary visual cortex, and do not contain any particularly sophisticated representation of shape, nor do they possess any explicit mechanism designed to tolerate image variation (e.g. from variation in view, lighting, etc.).

Pinto and colleagues previously described two *V1-Like* representations: *V1-Like* and *V1-like+*; code for both representations is available upon request. In the "default" *V1-like* representation, each input image is first resized by bicubic interpolation (the largest edge is resized to 150 pixels while preserving the aspect ratio), before conversion to grayscale and normalization to zero-mean and unit-variance. Each element in the output representation correspond to the "activity" of a simulated V1-simple-cell-like unit. Each response is computed by:

1. first locally normalizing the image (dividing each pixel's intensity value by the norm of the pixels in the 3x3 neighboring region),

2. applying a set of 96 spatially local (43x43 pixels) Gabor wavelets to the image (with a one pixel stride),

3. and normalizing the output values (dividing by the norm of the output values of all 3x3 spatial region across all Gabor filter types);

4. output values are finally thresholded (values below zero were clipped to zero) and clipped (values above one were clipped).

The 96 Gabors were chosen such that they spanned an exhaustive cross of 16 orientations (evenly spaced "around the clock") and 6 spatial frequencies (1/2, 1/3, 1/4, 1/6, 1/11, 1/18, 1/23, 1/35 cycles/pixel). The *V1-Like+* representation includes all of the *V1-Like* features, plus a grab-bag of easily-computed additional features (e.g. color and output histograms, see [10]).

In this study, we refer to the original versions of these representations as V1-like(A) and V1-like(A)+ and describe six new instances, as follows.

- Both V1-like(B) and V1-like(B)+ resize the largest edge of their input images by 75 pixels instead of 150. V1-like(B)+ concatenates 37x37 raw grayscale pixels to the feature vector instead of 75x75 (see [11]). Other parameters are unchanged from (A);

- V1-like(C) and V1-like(C)+ use slightly bigger Gabor filters (63x63 instead of 43x43) and cover an enlarged panel of 8 spatial frequencies (1/2, 1/3, 1/4, 1/6, 1/11, 1/18, 1/23, 1/35 cycles/pixel), for a total of 128 Gabor filters). Their output stack is downsampled to 10x10x128 with a 21x21 box-car filter instead of the original 30x30x96 with a 17x17 filter. The other parameters are unchanged from (A);

- V1-like(D) and V1-like(D)+ use much larger Gabor filters (125x125 instead of 43x43), and cover an enlarged panel of 24 spatial frequencies (1/2, 1/5, 1/8, 1/11, 1/14, 1/18, 1/22, 1/27, 1/31, 1/36, 1/41, 1/46, 1/52, 1/58, 1/64, 1/70, 1/76, 1/82, 1/89, 1/96, 1/103, 1/110, 1/117, 1/125) and 36 orientations (equally spaced "around the clock"), for a total of 864 Gabor filters. Their output stack is downsampled to 10x10x864 with a 21x21 box-car filter instead of the original 30x30x96 with a 17x17 filter. The other parameters are unchanged from (A);

These variants represent modest departures from the original *V1-Like* representations described in [11]. Since MKL-based blends benefit from the inclusion of as much diversity as possible, the use of these variants represents just a first step in optimization of the use of the V1-like representation class.

2.2. Classification by Optimally Combining Kernels

The classification of face images was performed using multi kernel learning (MKL) associated with a support vector machine (SVM). MKL allows the practitioner to optimize jointly over a convex linear combination of p kernels $K^* = \sum_{k=1}^{p} \beta_k K_k$ and the SVM parameters $\alpha \in \mathbb{R}^n$ and $b \in \mathbb{R}$, where n is the number of training examples. The value of the coefficients β, α and b are obtained by solving the following optimization problem:

$$\begin{cases} \min\limits_{\beta,\alpha,b} & \frac{1}{2}\left(\sum\limits_{k=1}^{p} \beta_k \alpha^T K_k \alpha\right) + C\sum\limits_{i=1}^{n} \xi_i \\ \text{s.t.} & \sum\limits_{k=1}^{p} \beta_k = 1 \quad \text{and} \quad \beta_k \geq 0 \quad \forall k \\ \text{with} & \xi_i = \max(0, 1 - y_i(\sum_{k=1}^{p} \beta_k K_k(x_i)^T \alpha + b)) \end{cases}$$

Where y_i is the binary label $\in \{-1, +1\}$ associated with the i-th training example x_i.

We solve this problem using the semi-infinite linear problem (SILP) formulation described in [18]. The implementation was taken "out-of-the-box" from the shogun-toolbox [31]. The combined kernels were all linear and were obtained after sphering the data – e.g. features were made to be zero-mean and unit-variance, with sphering parameters being estimated from the training examples. To avoid the MKL optimization unduely favoring any one kernel during training, their traces were normalized to one (i.e. by dividing each element of the training and testing matrix by the sum of the training matrix diagonal).

The SVM's regularization parameter C was fixed to 10^4 for all experiments. All the other parameters were set to their default values (see [31] for more details).

A full discussion of MKL methods is outside of the scope of the present paper, and is well covered elsewhere [1, 18, 13]. For the purpose of this work, MKL methods simply represent an expedient and powerful means to more fully exploit a large collection of features.

3. Experiments

3.1. Labeled Faces in the Wild Set

We first conducted experiments on the recent "Labeled Faces in the Wild" (LFW) face set (using the "View 2" subset from the LFW "funneled" version, see [7] for details). This set contains 13,233 images (250x250 pixels) of 5,749 individuals (see Figure 1 for examples) and was created to study the problem of face pair matching in unconstrained environments (i.e., given two face images, decide if they are from the same person or not). At a surface level, face images from the LFW set appear to be quite varied in appearance, and this is hailed as one of set's primary advantages.

3.1.1 Pair Matching

In this pair-matching setting, each representation variant described in Section 2 (i.e. each of the six variants for the *Pixels* representation and eight variants for the *V1-Like* representation) was used to produce six linear kernels as follows.

- The first kernel was the same as in [11] where the feature mapping is the element-wise squared difference of

(a) Examples of one individual from LFW.

(b) Examples of "same" and "different" pair of faces in LFW.

Figure 1. Examples taken from the "Labeled Faces in the Wild" (LFW) test set [24].

the representation outputs computed on a given pair of 250x250 images.

- The second and third kernels were also computed from 250x250 images but using an absolute-value difference or a square-root absolute-value difference respectively.

- The last kernels were computed using these three different element-wise differences (i.e. squared, absolute-value and square-root absolute-value) on 150x150 pair of images (cropped from the center).

Finally, for each training pair, the resulting feature vector was labeled as "same" or "different," and the task of labeling new (test) examples was treated as a two-category classification problem (theoretical chance being 50%). We followed the standard procedure described in [7, 24] and we report the mean classification accuracy ± s.e.m. computed from the ten random folds of 5,400 training and 600 testing examples from the "View 2" portion of the full LFW set.

	Grayscale	Color
no blur	66.02%±0.53	**68.33%±0.50**
Gaussian blur($\sigma = 1$)	66.12%±0.54	67.47%±0.53
Gaussian blur($\sigma = 2$)	66.12%±0.55	66.45%±0.64
All variants	**68.22%±0.41**	

Table 1. Performance on the "Labeled Faces in the Wild" (LFW) set using multiple-kernel learning (MKL) with kernels computed from the *Pixels* representations. The score of each cell is the result of the optimal combination of six kernels (see methods). All the variants add up to 36 kernels. Note that using all kernels doesn't improve performance significantly over the optimal blend of non-blurred color images.

	V1-like	V1-like+
Variant (A)	76.55%±0.49	78.52%±0.49
Variant (B)	73.23%±0.57	76.16%±0.56
Variant (C)	74.65%±0.38	77.30%±0.62
Variant (D)	73.43%±0.36	75.78%±0.49
All variants	**79.35%±0.55**	

Table 2. Performance on LFW set using MKL with kernels computed from the *V1-Like* representations. The score of each cell is the result of the optimal combination of 6 kernels (see methods). All the variants add up to 48 kernels. Note that using all kernels, our approach can get close to 80% accuracy.

3.1.2 Results

Table 1 summarizes the performance using MKL to combine variants of the *Pixels* baseline model. The best performance achieved is 68.33%±0.50 correct, using non-blurred color images. This is substantially more than theoretical chance (50%). More importantly, already this simple pixel-based approach outperforms some previously reported methods (e.g. see [21] for details).

The recognition accuracy of the *V1-Like* model variants is presented in Table 2, and a corresponding ROC curve is shown in Figure 2. Interestingly, an MKL blend of only six *V1-like(A)+* kernels (i.e., the representation taken, without modification, from [11]) scored 78.52%±0.49, which is not significantly different from the current state-of-the-art [21].

When all 48 *V1-Like* kernels were blended, performance reached 79.35%±0.55, establishing a new record (as of the time of writing of this manuscript) on this test set. Combining all 36 *Pixels* and 48 *V1-Like* kernels did not improve performance further.

Reference	Methods	Performance
Huang08 [6]	Nowak [9]	73.93%±0.49
	MERL	70.52%±0.60
	Nowak+MERL	76.18%±0.58
Wolf08 [21]	descriptor-based	70.62%±0.57
	one-shot-learning*	76.53%±0.54
	hybrid*	78.47%±0.51
This paper	Pixels/MKL	68.22%±0.41
	V1-like/MKL	**79.35%±0.55**

Table 3. Average performance comparison with the current state-of-the-art on LFW. *note that the "one-shot-learning" and "hybrid" methods from [21] cannot directly be compared to ours as they exploit the fact that individuals in the training and testing sets are mutually exclusive (i.e. using this property, you can build a powerful one-shot-learning classifier knowing that each test example is *different* from all the training examples, see [21] for more details. Our decision not to use such techniques effectively handicaps our results relative to reports that use themr).

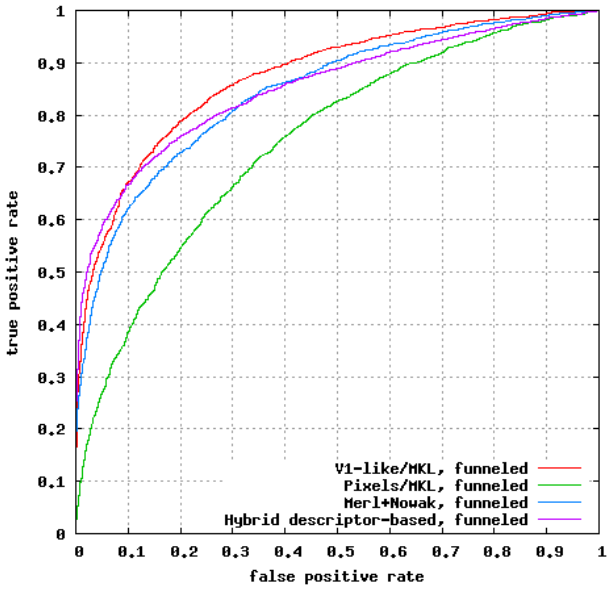

Figure 2. ROC curve comparison with the current state-of-the art on LFW. These curves were generated using the standard procedure described in [24].

3.2. Synthetic Face Set

At this point, we have shown that a combination of MKL techniques with previously described "trivial" feature representations is able to yield record levels of performance on a standard face recognition test set. However, this high level of performance could be due one of to two possible causes: 1) the powerful MKL back-end could be extracting a sophisticated, robust solution to face recognition from the relatively unsophisticated "parts" provided by the V1-like representation, or 2) the LFW set itself could contain more low-level regularities than previously appreciated, which the MKL-based back-end is more adept at exploiting.

To investigate whether the large-scale combination of kernels based on *Pixels* or *V1-Like* representations represents a robust solution to the face recognition problem, we conducted experiments using an ostensibly simpler parametric face set described in [11], using the a similar protocol as described in that work. Briefly, the image set consisted of two individual 3D faces meshes (one male, one female generated using the FaceGen software package [29]), rendered using the POV-Ray raytracing package [30] (see Figure 3 for examples). Because this image set only contains two individuals, it is arguably simpler than most other face recognition sets, which typically contain many individuals (e.g. almost 6,000, in the case of the LFW set). Critically, however, these synthetic faces were rendered with parametrically increasing amounts of variation in rotation, 2D position, and size, so that the performance of a system can be assessed as a function of the amount of variation present in the set. Here, as above, we used MKL-based classifiers, with a combination of kernels from the six *Pixels* representation variants and the eight *V1-Like* variants (see Materials and Methods). Test sets corresponding to seven levels of increasing variation (see Figure 3, x-axis labels) were created. For each level of variation, classifiers were trained with 150 randomly generated faces per individual and were tested using 150 examples.

Figure 3 shows the performance of the MKL combinations of the Pixels and V1-like baseline models with this synthetic set, as a function of the amount of parametric image variation. (i.e. position, viewpoint, scale, etc.). Echoing the results of [11], performance degrades rapidly as a function of image variation, with even modest amounts of variation resulting in chance performance. Interestingly, performance falls to a level statistically indistinguishable from chance at the same variation level as in [11] (the fourth data point in 3) and the use of a powerful large-scale classifier back-end does not rescue performance at this level. While the addition of an MKL back-end did produce some gains at smaller levels of image variation relative to that reported in [11] (e.g. the second and third points in Figure 3), it is clear that an MKL-based classifier built atop these simple features does not represent a particularly robust solution to the problem of unconstrained face recognition.

4. Discussion

In this study, we combined variants of the *Pixels* and *V1-Like* baseline models [11, 10] using a large-scale statistical learning tool ("out-of-the-box" MKL, [18]) to investigate how far you can get using only simple features. We presented evidence that this simple approach is capable of

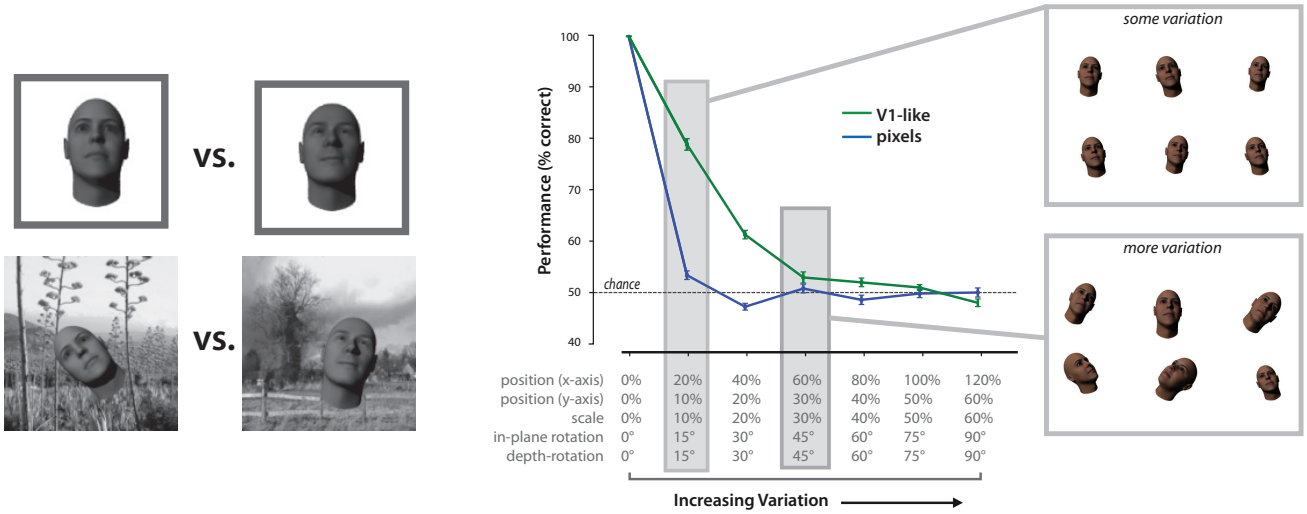

Figure 3. Performance of the *Pixels* and *V1-like* Representations with a MKL back-end on a synthetic face recognition task. Left top: examples of the faces to be discriminated in their default views, without any background (shown for illustration purposes); Left Bottom: examples of face images used here. The faces could appear in variety of sizes, positions, and orientations, and were randomly composited onto natural image backgrounds. For a human observer, this task is trivial, however even modest amounts of controlled view variation severely degrade performance of the MKL-backed *Pixels* and *V1-like* representations, confirming that these representations are not well suited for real-world face recognition, even with the addition of a more sophisticated back-end.

performing at a state-of-the-art level on the large "Labeled Faces in the Wild" (LFW) face recognition set, while failing on (an ostensibly simpler) synthetic set that includes more realistic view variation by design. Taken together, these results again urge for caution, as more sophisticated large scale kernel learning-based classifiers have the power to leverage good performance even from collections of relatively unsophisticated features. While it is still possible that this powerful machinery is building something "deeper" out of the simple parts provided to it, the extent of this sophistication is limited, at the very least. The MKL-backed system's inability to tolerate even modest amounts of variation (trivial for a human observer), raises the possibility that the MKL-backed system's gains on the LFW set may have more to do with extraction of low-level regularities than with progress towards the "core" problem.

4.1. The Importance of Good Benchmark Test Sets

These results underscore the importance of building test sets focussed on capturing the central computational challenges of real-world face and object recognition. The use of very large sets of "natural" images, while important, may not necessarily be optimal if used alone, as there is no clear way to ensure a realistic range of variation is present and there is no obvious way to control for undesired low-level regularities. A central concern with databases of "found" images from the internet is that photographers typically pose and frame their photos such that a limited range of views are highly over-represented. This effect may be further amplified by the manner in which the sets are assembled. For example, every face image included in the LFW set was the product of a successful detection by the Viola Jones algorithm applied to a set of pictures gathered from news articles on the Internet [7]. Even if the image diversity in LFW seems large, applying this face detector "filter" leads to an under-representation of lighting conditions and face views where the Viola-Jones detector does not excel (e.g. views from above, below or side; which can arguably be more challenging than frontal views). Obviously, such concerns are subject to practical trade-offs — though this automated procedure has biases, it enabled the authors to collect more than ten thousand images at a reasonable cost in terms of labor.

Large-scale methods are undoubtedly very powerful. However, this power represents a double-edged sword. On one hand, the use of large scale methods are now routinely responsible for the highest levels of performance in a variety of object and face recognition tests (e.g. [19, 2]). On the other hand, while such methods are adept at "wringing" substantial performance out of a test set and representation, there is no guarantee that such an exercise brings us closer to a real solution. Indeed, while large scale methods allowed us to achieve a high level of performance gains on the LFW set, we are unconvinced that these gains represent real progress. The cost of potentially false progress is magnified by the computational expense of large scale methods, which favor massive computational and memory footprints.

It is important to note that we are *not* claiming that

any previously reported result necessarily represents "false" progress. Previously reported methods may very well represent significant progress towards a solution. However, we argue that this progress will be difficult to see until, as a field, we are able to develop test sets that include realistic ranges of image variation. This will not be an easy task.

One approach that we advocate here is the complementary use of parametric, rendered image sets along with natural photographic sets. While synthetic sets have in some circles fallen out of favor, considered to be "toy" sets, our results here (along with previous reports [11, 10]) suggest that synthetic sets may in some ways be paradoxically more "natural" than a database of "found" photographic images, because they can span a realistic range of view, lighting, etc. variation, *by design*. In addition, because ground-truth is known, one can assess performance as a function of that variation. Finally, as computer graphics continue to become ever more realistic and accessible, the lines between natural and synthetic images are increasingly blurred, allowing a more natural interplay between both kinds of sets.

Of course, using synthetic images is not the only way to achieve controlled image variation. An alternative approach would be to use (or create) controlled photographic sets such as the PIE Face Set [16, 27] (or the NORB Object Set [8, 25]), which systematically vary parameters such as camera and lighting angle. However, while such sets have the appeal of being "real," it is extremely difficult and time-consuming to create a set that spans a sufficient number of axes of variation (i.e. six degrees of freedom in view, multiple light sources, different backgrounds, etc.), and failure to span enough axes results in an incomplete surrogate for the full range of variation in the real world. As a point of reference, for the PIE set, a simple unblended V1-like(A)+ already achieves 87.9%$\pm$0.3 performance [1], indicating that low-level regularities are likely nonetheless present. While a controlled photographic set with adequate variation is certainly theoretically possible, we are not aware of a set that meets this goal. Meanwhile, synthetic sets offer extreme practicality and flexibility.

4.2. New Baselines for Face Recognition

As previously argued in [11, 10], one function for low-level "baseline" models, such as the *V1-Like* model, is to set a baseline mark against which performance of other systems can be compared. Test sets where a "trivial" model performs well can still be highly useful, provided the level of performance of that "trivial" model is taken into account when evaluating performance, and provided that there is still "headroom" left with respect to the test set (i.e. the trivial model doesn't perform at 100%). That is, to be reassured that a purpose-built system is going beyond low-level regularities, the performance of the purpose-built vision system should ideally be substantially higher than the performance of a "trivial" model.

The nature of multiple kernel methods also opens up an additional avenue for integrating trivial baselines directly into the discovery process. In particular, if the simple *V1-like* representation presented here were added to the collection of representations under evaluation (i.e. including the purpose-built representation under study), then the *V1-like* representation can "soak up" some of the performance gains due to low-level regularities, making clearer what contributions are made by the purpose-built representation. In such a scenario, one would want the inclusion of the purpose-built representation to result in substantial improvement over the V1-like representation alone. To some extent, interpretation of the weights produced by the MKL approach [18, 17] could offer valuable insights into what contributions the purpose-built representation is making.

We are clearly not the first to identity the importance of evaluation in driving progress in face and object recognition [12]; our results add to a long-standing process of evaluation and re-evaluation of how algorithms and systems are evaluated. Going forward, large-scale techniques such as MKL will have an important role to play in face and object recognition, however, their use will also require redoubled efforts in collecting and creating test sets that properly channel and direct that power.

[1] 68-way one-against-all, chance is at 1.5%

5. Acknowledgements

We would like to thank Antonio Torralba and Ce Liu for encouraging this work, and NVIDIA Corporation for hardware support. This study was funding in part by The National Institutes of Health (NEI R01EY014970), The McKnight Endowment for Neuroscience, Dr. Gerald Burnett and Marjorie Burnett, and The Rowland Institute of Harvard.

References

[1] F. Bach, G. Lanckriet, and M. Jordan. Multiple kernel learning, conic duality, and the SMO algorithm. *Proceedings of the International Conference on Machine learning (ICML)*, 2004.

[2] A. Bosch, A. Zisserman, and X. Munoz. Representing shape with a spatial pyramid kernel. *Proceedings of the International Conference on Image and Video Retrieval (CIVR)*, 2007.

[3] B. Collins, J. Deng, L. Kai, and L. F.-F. L. Towards scalable dataset construction: An active learning approach. *Proceedings of the European Conference on Computer Vision (ECCV)*, 2008.

[4] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. *Work-*

shop on Generative-Model Based Vision in the Computer Vision and Pattern Recognition Conference (CVPR), 2004.

[5] G. Griffin, A. Holub, and P. Perona. The caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007.

[6] G. Huang, M. Jones, and E. Learned-Miller. LFW Results Using a Combined Nowak Plus MERL Recognizer. *Faces in Real-Life Images Workshop in European Conference on Computer Vision (ECCV)*, 2008.

[7] G. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. Technical report, 07-49, University of Massachusetts, Amherst, October 2007.

[8] Y. LeCun, F. Huang, and L. Bottou. Learning methods for generic object recognition with invariance to pose and lighting. *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, 2004.

[9] E. Nowak and F. Jurie. Learning visual similarity measures for comparing never seen objects. *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, 2007.

[10] N. Pinto, D. Cox, and J. DiCarlo. Why is Real-World Visual Object Recognition Hard. *PLoS Computational Biology*, 4(1):e27, 2008.

[11] N. Pinto, J. DiCarlo, and D. Cox. Establishing Good Benchmarks and Baselines for Face Recognition. *Faces in Real-Life Images Workshop in European Conference on Computer Vision (ECCV)*, 2008.

[12] J. Ponce, T. Berg, M. Everingham, D. Forsyth, M. Hebert, S. Lazebnik, M. Marszalek, C. Schmid, B. Russell, A. Torralba, et al. Dataset Issues in Object Recognition. *Lecture Notes in Computer Science*, 4170:29, 2006.

[13] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet. More efficiency in multiple kernel learning. 2007.

[14] B. Russell, A. Torralba, K. Murphy, and W. Freeman. LabelMe: A Database and Web-Based Tool for Image Annotation. *International Journal of Computer Vision*, 77(1):157–173, 2008.

[15] L. Shamir. Evaluation of Face Datasets as Tools for Assessing the Performance of Face Recognition Methods. *International Journal of Computer Vision*, 79(3):225–230, 2008.

[16] T. Sim, S. Baker, and M. Bsat. The CMU pose, illumination, and expression database. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2003.

[17] S. Sonnenburg, G. Ratsch, and C. Schafer. Learning Interpretable SVMs for Biological Sequence Classification. *Proceedings of the Regulatory Genomics and Systems Biology 2008 Conference (RECOMB)*, 2005.

[18] S. Sonnenburg, G. Rätsch, C. Schäfer, and B. Schölkopf. Large Scale Multiple Kernel Learning. *Journal of Machine Learning Research*, 7:1531–1565, 2006.

[19] M. Varma and D. Ray. Learning The Discriminative Power-Invariance Trade-Off. *Proceedings of the International Conference on Computer Vision (ICCV)*, 2007.

[20] L. von Ahn, R. Liu, and M. Blum. Peekaboom: a game for locating objects in images. *Proceedings of the International Conference on Human Factors in Computing Systems (SIGCHI)*, 2006.

[21] L. Wolf, T. Hassner, and Y. Taigman. Descriptor based methods in the wild. *Faces in Real-Life Images Workshop in European Conference on Computer Vision (ECCV)*, 2008.

[22] AR Face Set. http://cobweb.ecn.purdue.edu/ aleix/ar.html.

[23] CVL Face Set. http://www.lrv.fri.uni-lj.si/facedb.html.

[24] Labeled Faces in the Wild Set. http://vis-www.cs.umass.edu/lfw.

[25] NORB Object Set. http://www.cs.nyu.edu/ ylclab/data/norb-v1.0.

[26] ORL Face Set. http://www.cl.cam.ac.uk/Research/DTG/attarchive/facedatabase.html.

[27] PIE Face Set. http://www.ri.cmu.edu/projects/project_418.html.

[28] Yale Face Set. http://cvc.yale.edu.

[29] FaceGen, by Singular Inversions, Inc. http://www.facegen.com.

[30] POV-Ray: The Persistence of Vision Ray Tracer. http://www.povray.org.

[31] Shogun: A Large Scale Machine Learning Toolbox. http://www.shogun-toolbox.org.