Nonlinear Regression via Deep Negative Correlation Learning

Le Zhang, Zenglin Shi, Ming-Ming Cheng, Yun Liu, Jia-Wang Bian,
Joey Tianyi Zhou, Guoyan Zheng and Zeng Zeng

Abstract—Nonlinear regression has been extensively employed in many computer vision problems (e.g., crowd counting, age estimation, affective computing). Under the umbrella of deep learning, two common solutions exist i) transforming nonlinear regression to a robust loss function which is jointly optimizable with the deep convolutional network, and ii) utilizing ensemble of deep networks. Although some improved performance is achieved, the former may be lacking due to the intrinsic limitation of choosing a single hypothesis and the latter may suffer from much larger computational complexity. To cope with those issues, we propose to regress via an efficient "divide and conquer" manner. The core of our approach is the generalization of negative correlation learning that has been shown, both theoretically and empirically, to work well for non-deep regression problems. Without extra parameters, the proposed method controls the bias-variance-covariance trade-off systematically and usually yields a deep regression ensemble where each base model is both "accurate" and "diversified." Moreover, we show that each sub-problem in the proposed method has less Rademacher Complexity and thus is easier to optimize. Extensive experiments on several diverse and challenging tasks including crowd counting, personality analysis, age estimation, and image super-resolution demonstrate the superiority over challenging baselines as well as the versatility of the proposed method. The source code and trained models are available on our project page: https://mmcheng.net/dncl/.

Index Terms—deep learning, deep regression, negative correlation learning, convolutional neural network.

1 Introduction

We address regression problems which aim at analyzing the relationship between dependent variables (targets) and independent variables (inputs). Regression has been applied to a variety of computer vision problems including crowd counting [1], age estimation [2], affective computing [3], image super-resolution [4], visual tracking [5] and so on. Pioneering works within this area typically learn a mapping function from hand-crafted features (*e.g.*, Histogram of Oriented Gradients (HoG), Scale-Invariant Feature Transform (SIFT)) to the desired output (*e.g.*, ages, affective scores, density maps, and so on).

Recently, transforming a regression problem to an optimizable robust loss function jointly trained with deep Convolutional Neural Network (CNN) has been reported to be successful to some extent. Most of the existing deep learning based regression approaches optimize the L2 loss function together with a regularization term, where the goal is to minimize the mean square error between the network prediction and the ground-truth. However, it is well known that the mean square error is sensitive to outliers, which are essentially the samples that lie at an abnormal distance from other training samples in the objective space. In this case, samples that are rarely encountered in the training data may have a disproportionally high weight and consequently influence the training procedure by reducing the generalization ability. To this end, Ross Girshick [6] introduced a SmoothL1 loss for bounding box regression. As a special case of Huber loss [7], the SmoothL1 loss combines the concept of L2 loss and L1 loss. It behaves as an L1 loss when the absolute value of the error is high and switches back to L2 loss when the absolute value of the error is close to zero. Besides, Belagiannis *et al.* [8] propose a deep regression network that achieves robustness to outliers by minimizing Tukey's biweight function [9], [10] .

While tremendous progress has been achieved later by employing robust statistical estimations together with specially-designed network architecture to explicitly address outliers, they may fail to generalize well in practice. As studied in [11], a single model may be lacking due to the statistical, computational and representational limitations. To this end, a great deal of research has gone into designing multiple regression systems [11]–[14]. However, existing methods for ensemble CNNs [15]–[17] typically trained multiple CNNs, which usually led to much larger computational complexity and hardware consumption. Thus, these ensemble CNNs are rarely used in practical systems.

In this paper, we propose a Deep Negative Correlation Learning (DNCL) approach which learns a pool of diversified regressors in a "divide and conquer" manner. Each regressor is jointly optimized with CNNs by an amended cost function, which penalizes correlations with others. Our approach inherits the advantage of traditional *Negative Correlation Learning (NCL)* [18], [19] approaches, that systematically controls the trade-offs among the *bias-variance-covariance* in the ensemble. Firstly, by dividing the task into multiple "negatively-correlated" sub-problems, the proposed method shares the essence of ensemble learning and yield more robust estimations than a single network [13], [19].

- *The first two authors are the joint first author, and MM Cheng is the corresponding author.*
- *L Zhang, JT Zhou, and Z Zeng are with the Agency for Science Technology and Research (A*STAR), Singapore.*
- *Z Shi is with the University of Amsterdam.*
- *G Zheng is with the School of Biomedical Engineering, Shanghai Jiaotong University, China.*
- *Ming-Ming Cheng, Yun Liu are with the TKLNDST, College of Computer Science, Nankai University, China.*
- *JW Bian is with the School of Computer Science, The University of Adelaide, Adelaide, Australia*

Manuscript received April 19, 2005; revised August 26, 2015.

Secondly, thanks to the rich feature hierarchies in deep networks, each sub-problem could be solved by a feature subset. In this way, the proposed method has a similar amount of parameters with a single network and thus is much more efficient than most existing deep ensemble learning [15]–[17]. Simplicity and efficiency are central to our design. The proposed methods are almost complementary to other advanced strategies for individual regression tasks.

A preliminary version of this work was presented in CVPR 2018 [1], which provides an application of DNCL for crowd counting. This paper adds to the initial version in the following aspects:

- We provide more theoretical insights on the Rademacher complexity.
- We extend the original work to deal with more regression-based problems, which allows the use of state-of-the-art network structures that give an important boost to performance for the proposed method.
- More comprehensive literature review, considerable new analysis and intuitive explanations are added to the initial results.

2 Related Work

2.1 Regression

We first briefly introduce the commonly used loss function for regression based deep learning computer vision tasks, followed by summarizing the existing ensemble regression techniques.

Deep Regression. Recently, learning a mapping function to predict a set of interdependent continuous values by deep networks is popular. One example could be object detection where the target is to regress the bounding box for precise localization [20]. Other examples include regressing the facial points in facial landmark detection [21] and positions of the body in human pose estimation [22]. The L2 loss function is a natural choice for solving such problems. Zhang *et al.* [23] further utilized L2 regularization to increase the robustness of network for both landmark detection and attribute classification. Similar strategies were also applied in object detection [24].

The commonly used L2 loss in regression problems may not generalize well in the case of outliers because outliers can have a disproportionally high weight, and consequently influence the training procedure by reducing the generalization ability and increasing the convergence time. To this end, a SmoothL1 loss [6] was reported to be more robust than L2 loss when outliers are present in the dataset:

$$\mathrm{Smooth}_{\mathrm{L}_1}(\xi) = \begin{cases} 0.5\xi^2 & \text{if } |\xi| < 1 \\ |\xi| - 0.5 & \text{otherwise} \end{cases}, \tag{1}$$

where ξ stands for the prediction error. Motivated by the recent success in robust statistics [9], an M-estimator based [10] loss function, called Tukey Loss [8], was proposed for both human pose estimation and age estimation [8]. More specifically,

$$\mathrm{Tukey}(\hat{\xi}) = \begin{cases} \frac{c^2}{6}[1-(1-(\frac{\hat{\xi}}{c})^2)^3] & \text{if } |\hat{\xi}| \le c \\ \frac{c^2}{6} & \text{otherwise} \end{cases}, \tag{2}$$

where c is a tuning parameter, and is commonly set to 4.6851, which gives approximately 95% asymptotic efficiency as L2 minimization on the standard normal distribution of residuals. $\hat{\xi}$ is a scaled version of the residual ξ by computing the median absolute deviation by:

$$\begin{aligned} \hat{\xi} &= \frac{\mathbf{y} - \bar{\mathbf{y}}}{1.4826 \times MAD}, \\ \mathrm{MAD} &= \underset{i\in\{1,\cdot,N\}}{\mathrm{median}}(|\xi_i - \underset{k\in\{1,\cdot,N\}}{\mathrm{median}}(\xi_k)|), \end{aligned} \tag{3}$$

where $\mathbf{y}$, $\bar{\mathbf{y}}$ and N stands for the ground-truth label, predicted result and number of data samples, respectively. In case of regressing multiple outputs, the MAD values are calculated independently.

Our proposed DNCL method could also be regarded as a loss function, which is readily pluggable into existing CNN architecture and amenable to training via backpropagation. Without extra parameters, the proposed methods mimic ensemble learning and have better control of the trade-off between the intrinsic bias, variance, and co-variance. We evaluate the proposed method on multiple challenging and diversified regression tasks. When combined with the state-of-the-art network structure, our method could give an important boost to the performance of existing loss functions mentioned above.

Ensemble Regression. Ensemble methods are wildly regarded to be better than single model if the ensemble is both "accurate" and "diversified" [11]–[13]. As studied in [11], a single model was less generalizable from the statistical, computational and representational point of view. To this end, a bunch of research has gone into designing multiple regression systems. For instance, the accuracy and diversity in a typical decision tree ensemble [5], [12], [14], [25] were guaranteed by allowing each decision tree grow to its maximum depth and utilizing feature subspace, respectively. Boosting [26] generated a new regressor with an amended loss function based on the loss of the existing ensemble models.

Motivated by the success of ensemble methods, several deep regression ensemble methods were proposed as well. However, existing methods for training CNN ensembles [15], [17] usually generated multiple CNNs separately. In this case, the resulting system usually yielded a much larger computational complexity compared with single models and thus was usually very slow in terms of both training and inference, which naturally limited their applicability for resource-constrained scenarios.

One of the exceptions could be the Deep Regression Forest (DRF) [27] which reformulated the split nodes as a fully connected layer of a CNN and learned the parameter of CNN and tree nodes jointly by an alternating strategy. Firstly, by fixing the leaf nodes, the internal nodes of trees as well as the CNN were optimized by back-propagation. After that, both the CNN and the internal nodes were frozen, and the leaf nodes were learned by iterating a step-size free and fast converging update rule derived from Variational Bounding. We show the proposed method could also be combined with the concept of DRF. The resulting system is much simpler to learn and yield a significant improvement enhancement, as elaborated in Section 4.3.

2.2 Applications

The proposed method is generic and could be applied to a wide range of regression tasks. It mimics ensemble learning without extra parameters and helps to learn more generalizable features through better control of the trade-off between the intrinsic bias, variance, and co-variance. We evaluated it on multiple challenging and diversified regression tasks including crowd counting,

age estimation, personality analysis and image super-resolution. Simplicity is central to our design and the strategies adopted in the proposed method are complementary to many other specially-designed techniques for each task. When combined with state-of-the-art network structure for each task, our proposed method is able to yield an important boost to the baseline methods. Below we provide a detailed review on the recent advances in each task.

Crowd Counting. Counting by regression is perceived as the state-of-the-art at present. The regression-based methods have been widely studied and reported to be computationally feasible with modern hardware, robust with parameters and accurate across various challenging scenarios. A deep CNN [28] was trained alternatively with two related learning objectives, crowd density classification and crowd counting. However, it relied heavily on a switchable learning approach and was not clear how these two objective functions can alternatively assist each other. Wang *et al.* [29] proposed to directly regress the total people number by adopting AlexNet [30], which has now been found to be worse than the methods regressing density map. This observation suggests that reasoning with rich spatial layout information from convolutional feature maps is necessary. Boominathan *et al.* [31] proposed a framework consisting of both deep and shallow networks for crowd counting. It was reported to be more robust with scale variations, which have been addressed explicitly by other studies [32]–[34] as well. Switching CNN was introduced in [35], where patches from a grid within a crowd scene were relayed to independent CNN regressors based on crowd count prediction quality of the CNN established during training. Arteta *et al.* [36] augmented and interleaved density estimation with foreground-background segmentation and explicit local uncertainty estimation under a new deep multi-task architecture. Noroozi *et al.* [37] used counting as a pretext task to train a neural network with a contrastive loss and showed improved results on transfer learning benchmarks.

Personality Analysis. Recent personality-related work with visual cues attempted to identify personality from body movement [38], facial expression change [39], [40], combining acoustic cues [41], eye gaze [42], and so on. In addition, recognizing personality traits using deep learning on images or videos has also been extensively studied. 'ChaLearn 2016 Apparent Personality Analysis competition' [3] provided an excellent platform, where researchers could assess their deep models on a large annotated big-five personality traits dataset. Instead of classifying pre-defined personality categories, common practices use a finer-grained representation, in which personalities are distributed in a five-dimensional space spanned by the dimensions of *Extraversion, Agreeableness, Conscientiousness, Neuroticism, and Openness* [3]. This is advantageous in the sense that personality states can be represented at any level of the aforementioned big-five personality traits. A Deep Bimodal Regression framework based on both video and audio input was utilized in [43] to identify personality. A similar work from Güçlütürk *et al.* [44] introduced a deep audio-visual residual network for multimodal personality trait recognition. In addition, a volumetric convolution and Long-Short-Term-Memory (LSTM) based network was introduced by Subramaniam *et al.* [45] for learning audio-visual temporal patterns. A pre-trained CNN was employed by Gürpınar *et al.* [46] to extract facial expressions as well as ambient information for personality analysis. For more related work on personality analysis, please refer to recent surveys [47], [48].

Age Estimation. Age estimation from face images is gaining its popularity since the pioneering work of [49]. Conventional regression methods include but are not limited to kernel method [50], [51], hierarchical regression [52], randomized trees [53], label distribution [54], and so on. Recently, end-to-end learning with CNN has also been widely studied for age estimation. Ni *et al.* [55] firstly proposed a four layer CNN for age estimation. Niu *et al.* [56] reformulated age estimation as an ordinal regression problem by using end-to-end deep learning methods. In particular, age estimation in their setting was transformed into a series of binary classification sub-problems. Ranking CNN was introduced in [57], where each base CNN was trained with ordinal age labels. In [58], anchored Regression Networks were introduced as a smoothed relaxation of a piece-wise linear regressor for age estimation through the combination of multiple linear regressors over soft assignments to anchor points. Li *et al.* [59] designed a Deep Cross-Population (DCP) age estimation model with a two-stage training strategy, in which a novel cost-sensitive multitask loss function was first used to learn transferable aging features by training on the source population. Then, a novel order-preserving pair-wise loss function was utilized to align the aging features of the two populations. DEX [60] solved age estimation by way of deep classification followed by a softmax expected value refinement. Shen *et al.* [27] extended the idea of a randomized forest into deep scenarios and show remarkable performances for age estimation.

Single Image Super-resolution. With the far-reaching application in medical imaging, satellite imaging, security, and surveillance, single image super-resolution is a classic computer vision problem, which aims to recover a high resolution (HR) image from a low-resolution (LR) image. Since the use of fully convolutional networks for super-resolution [61], many advanced deep architectures have been proposed. For instance, Cascaded Sparse Coding Network (CSCN) [29] combined the strengths of sparse coding and deep network. An efficient sub-pixel convolution layer was introduced in [62] to better upscale the final LR feature maps into the HR output. A PCA-inspired collaborative representation cascade was introduced in [63]. A novel residual dense network (RDN) was designed [64] to fully exploit the hierarchical features from all the convolutional layers. Specifically, they proposed residual dense block (RDB) to extract abundant local features via dense connected convolutional layer. A deeply-recursive convolutional network (DRCN) was proposed in [65], which increased the network depth by a recursive layer without introducing new parameters for additional convolutions. A very deep fully convolutional encoding-decoding framework was employed in [66] to combine convolution and deconvolution. Wei *et al.* [67] reformulated image super-resolution as a single-state recurrent neural network (RNN) with finite unfoldings and further designed a dual-state design, the Dual-State Recurrent Network (DSRN). Deep Back-Projection Networks (DBPN) [68] exploited iterative up- and downsampling layers to provide an error feedback message for projection errors at each stage. In [69], a residual in residual (RIR) structure was introduced. The concept of non-local

learning was adopted in [70] for image super-resolution. For more research work, please refer to [71].

3 Proposed Method

3.1 Background

Before elaborating the proposed regression method, we first briefly present the notations and background knowledge. We assume that we have access to N training samples, $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$. The samples are M dimensional: $X \in \mathcal{X} \subseteq I\!R^M, M = H \times W \times C$, where H, W, and C denote the height, width, and number of channels of input image respectively. Our objective is to predict their regression labels, *i.e.*, $Y = \{y_1, \ldots, y_N\}$, where $Y \in \mathcal{Y} \subseteq I\!R^{\mathcal{M}}$. We denote a generic data point by $\mathbf{x}$ and use $\mathbf{x}_\diamond$, with $\diamond$ denoting the place-holder for the index wherever necessary. Similarly, we use M and $\mathcal{M}$ to represent the dimensionality of a generic input data and its label, respectively. We achieve our goal by learning a mapping function $G : \mathcal{X} \rightarrow \mathcal{Y}$, where $G \in \mathcal{G}$.

The learning problem is to use the set X to learn a mapping function G, parameterized by θ, to approximate their label Y as accurate as possible:

$$L(G) = \int (G(X, \theta) - Y)^2 p(X, Y) d(X, Y), \tag{4}$$

In practice, as data distribution $p(X, Y)$ is unknown, Eqn. (4) is usually approximated by

$$L(G) = \frac{1}{N} \sum_{i=1}^{N} (G(\mathbf{x}_i, \theta) - y_i)^2. \tag{5}$$

We omit the input and parameter vectors. Without ambiguity, instead of $G(X, \theta)$, we write simply G. We use the shorthand expectation operator E to represent the generalization ability on testing data. *Bias-variance decomposition* [19] theorem states that the regression error of a predictor can be decomposed into its *bias* $\mathbb{B}$ and *variance* $\mathbb{V}$:

$$E[(G - Y)^2] = \underbrace{(E[G] - Y)^2}_{\mathbb{B}(G)^2} + \underbrace{E[(G - E[G])^2]}_{\mathbb{V}(G)}. \tag{6}$$

It is a property of the generalization error in which bias and variance have to be balanced against each other for best performance.

A single model, however, turns out to be far from optimal in practice which has been evidenced by several studies, both theoretically [13], [19] and empirically [72], [73]. Consider the ensemble output $\tilde{G}$ by averaging individual's response G_k, *i.e.*,

$$\tilde{G} = \frac{1}{K} \sum_{k=1}^{K} G_k. \tag{7}$$

Here we restrict our analysis to the uniform combination case which is commonly used in practice, although the decomposition presented below generalize to non-uniformly weighted ensembles as well. Posing the ensemble as a single learning unit, its bias-variance decomposition can be shown by the following equation:

$$E[(\tilde{G} - Y)^2] = \underbrace{(E[\tilde{G}] - Y)^2}_{\mathbb{B}(\tilde{G})^2} + \underbrace{E[(\tilde{G} - E[\tilde{G}])^2]}_{\mathbb{V}(\tilde{G})} \tag{8}$$

Consider ensemble output in Eqn. (7), it is straightforward to show:

$$\begin{aligned} E[(\tilde{G} - Y)^2] = &\left(\frac{1}{K} \sum_{k=1}^{K} \underbrace{(E[G_k] - Y)}_{\mathbb{B}(G_k)} \right)^2 \\ &+ \frac{1}{K^2} \sum_{k=1}^{K} \underbrace{E[(G_k - E[G_k])^2]}_{\mathbb{V}(G_k)} \\ &+ \frac{1}{K^2} \sum_{k=1}^{K} \sum_{j \neq k} \underbrace{E[(G_k - E[G_k])(G_j - E[G_j])]}_{\mathbb{C}(G_k, G_j)}, \end{aligned} \tag{9}$$

where $\mathbb{C}$ denotes for *covariance*.

The *bias-variance-covariance* decomposition in Eqn. (9) illustrates that, in addition to the internal bias and variance, the generalization error of an ensemble depends on the covariance between the individuals as well.

It is natural to show

$$\begin{aligned} \frac{1}{K} \sum_{k=1}^{K} E[(G_k - Y)^2] = &\mathbb{B}(G)^2 + [K \times \mathbb{V}(G) \\ &+ \frac{1}{K} \sum_{k=1}^{K} (E[G_k] - E[\tilde{G}])^2] \end{aligned}$$

and

$$\begin{aligned} \frac{1}{K} \sum_{k=1}^{K} E[(G_k - \tilde{G})^2] = &- [\mathbb{V}(G) + \mathbb{C}(G)] + [K \times \mathbb{V}(G) \\ &+ \frac{1}{K} \sum_{k=1}^{K} (E[G_k] - E[\tilde{G}])^2]. \end{aligned}$$

Then it is easy to show

$$\begin{aligned} E[(\tilde{G} - Y)^2] = &\frac{1}{K} \sum_{k=1}^{K} E[(G_k - Y)^2] \\ &- \frac{1}{K} \sum_{k=1}^{K} E[(G_k - \tilde{G})^2]. \end{aligned} \tag{10}$$

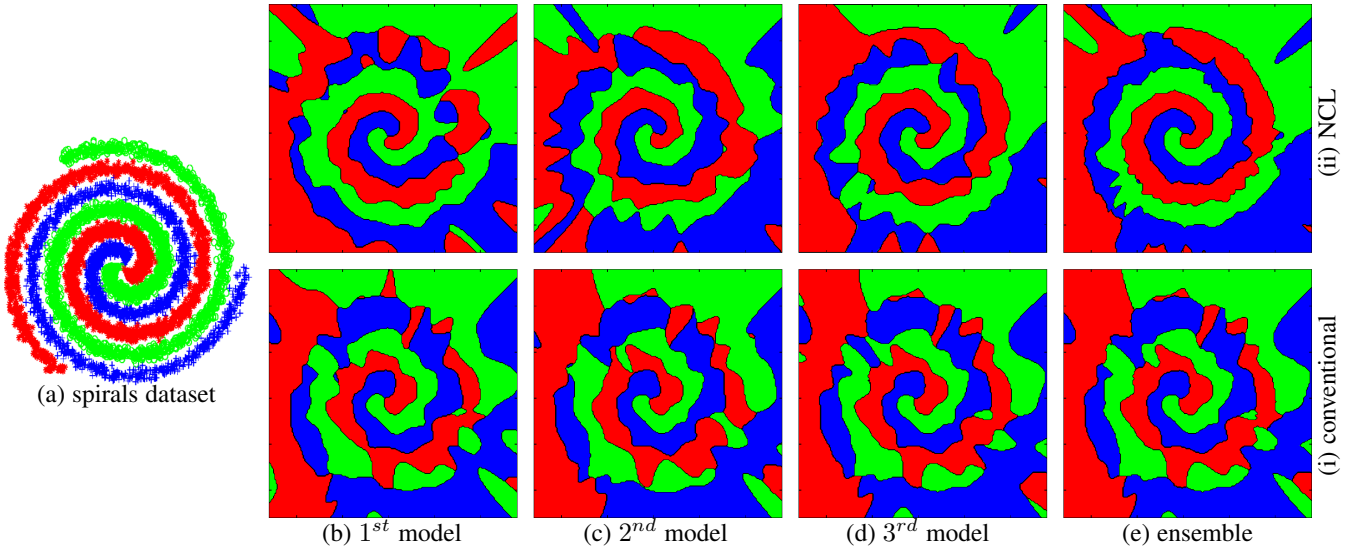
Eqn. (10) explains the effect of error correlations in an ensemble model by stating that *the quadratic error of the ensemble estimator is guaranteed to be less than or equal to the average quadratic error of the component estimators*. This is also in line with the strength-correlation theory [12], which advocates learning a set of both accurate and decorrelated models.

3.2 Deep Negative Correlation Learning

3.2.1 Our Method

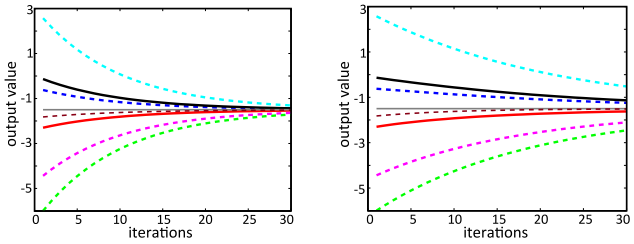
Conventional ensemble learning methods such as bagging [74] and Random Forest [12] train multiple models independently. This may not be optimal because, as demonstrated in Eqn. (10), the ensemble error consists of both the individual error and the interactions within the ensemble. Based on this, we proposed a "divide and conquer" deep learning approach by learning a correlation regularized ensemble on top of deep networks with the following objective:

$$\begin{aligned} L_k &= \frac{1}{2}(G_k - Y)^2 + \lambda (G_k - \tilde{G}) (\sum_{i \neq k} (G_i - \tilde{G})), \\ &= \frac{1}{2}(G_k - Y)^2 - \lambda (G_k - \tilde{G})^2, \end{aligned} \tag{11}$$



(a) spirals dataset (b) 1^{st} model (c) 2^{nd} model (d) 3^{rd} model (e) ensemble

Fig. 1. Decision surfaces for classification of artificial spirals dataset for both (i) conventional ensemble learning and (ii) NCL learning. The shading of the background shows the decision surface for that particular class. The upper part of the figure corresponds to NCL learning and lower part stands for conventional ensemble learning. (b)-(d) shows the decision surface of individual model and (e) shows the ensemble decision surface arising from averaging over its individual models. NCL in (b)-(e) leads to much diversified decision surface where errors from individual models may cancel out thus resulting in much better generalization ability. Best viewed in color.



(a) Conventional ensemble learning (b) NCL

Fig. 2. Demonstration of the training process of conventional ensemble learning and NCL. The central solid gray line represents the ground truth and all other lines stand for different base models. Although both conventional ensemble learning (left part) and NCL (right part) may lead to correct estimations by simple model averaging, NCL results in much diversified individual models which make error cancellation possible on testing data.

More specifically, we consider our mapping function as an ensemble of predictors as defined in Eqn. (7) where each base predictor is posed as:

$$G_k(\mathbf{x}_i) = G_k^{\mathcal{Q}}(G_k^{\mathcal{Q}-1}\cdots(G_k^1(\mathbf{x}_i)))), \quad k = 1, 2\cdots K, i = 1, 2\cdots N \tag{12}$$

where k, i, and $\mathcal{Q}$ stand for the index for individual models, the index for data samples and the depth of the network, respectively. More specifically, each predictor in the ensemble consists of cascades of feature extractors G_k^q, $q = 1, 2\cdots\mathcal{Q}-1$ and regressor $G_k^{\mathcal{Q}}$. Motivated by the recent success of CNNs on visual recognition tasks, each feature extractor G_k^q is embodied by a typical layer of a CNN. Below we present the details for each task.

3.2.2 Network Structure

The proposed method can be efficiently encapsulated into existing deep CNN thanks to its rich feature hierarchy. In our implementation, as illustrated in Fig. 3, lower levels of feature extractors are shared by each predictor for efficiency, *i.e.*, $G_k^q = G^q$, $q = 1, 2\cdots, \mathcal{Q}-1$, $k = 1, 2\cdots, K$. Inspired by the subspace idea in ensemble learning [12], we divide the outputs of the highest level feature extractor $G_k^{\mathcal{Q}-1}$ to different subsets, each of which is used as input for different regressor $G_k^{\mathcal{Q}}$ to encourage diversities. This is implemented via the well-established *group convolution* strategy [30]. Each regressor is optimized by an amended cost function as defined in Eqn. (11). Generally speaking, network specification of G_k^q is problem-dependent, and we show that the proposed method is end-to-end-trainable and independent of the backbone network architectures.

Crowd counting. We employ a deep pretrained VGG16 network for this task and make several modifications. Firstly, the stride of the fourth max-pool layer is set to 1. Secondly, the fifth pooling layer was further removed. This provides us with a much larger feature map with richer information. To handle the receptive-field mismatch caused by the removal of stride in the fourth max-pool layer, we then double the receptive field of convolutional layers after the fourth max-pool layer by using the technique of *holes* introduced in [75]. We also include another variant of the proposed method called "NCL" which is a shallow network optimized with the same loss function. The details of this network will be elaborated in Section 4.5.

Personality analysis. We utilize a truncated 20 layer version of the *SphereFace* model [76] for personality analysis. We first detect and align faces for each input image with well-established *MTCNN* [77]. As we are dealing with videos, in order to speed up training and reduce the risk of over-fitting, we take a similar approach as done in [78] to first sparsely sample 10 frames from each video in a randomized manner. Average pooling is further used to aggregate multiple results for the same video.

Age estimation. For age estimation, we use the network

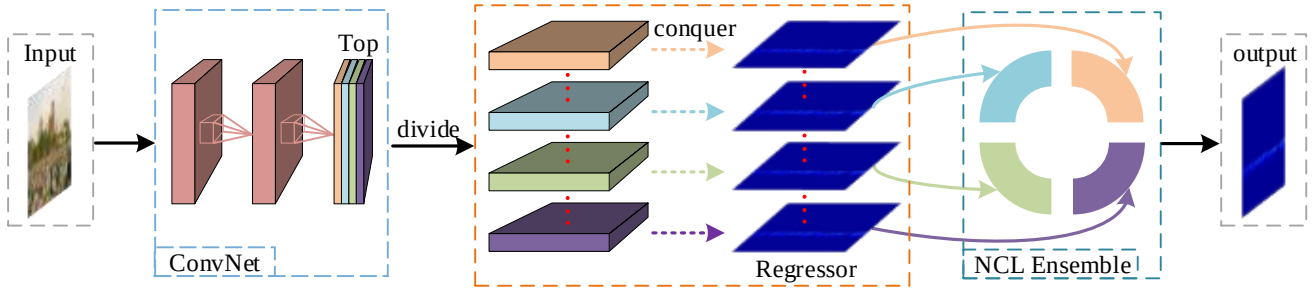


Fig. 3. Details of the proposed DNCL. Regression is formulated as ensemble learning with the same amount of parameter as a single CNN. DNCL processes the input by a stack of typical convolutional and pooling layers. Finally, a "divide and conquer" strategy is adapted to learn a pool of regressors to regress the output on top of each convolutional feature map at top layers. Each regressor is jointly optimized with the CNN by an amended cost function, which penalizes correlations with others to make better trade-offs among the bias-variance-covariance in the ensemble.

backbone of the deep forest [27]. It reformulates the split nodes of a decision forest as a fully connected layer of a CNN and learns both split nodes and leaf nodes in an iterative manner. More specifically, by fixing the leaf nodes, the split nodes as well as the CNN parameters are optimized by back-propagation. Then, by fixing the split nodes, the leaf nodes are optimized by iterating a step-size free and fast converging update rule derived from Variational Bounding. Instead of using this iterative strategy, we use the proposed NCL loss in each node to make them both accurate and diversified.

Image super-resolution. For image super-resolution, we choose the state-of-the-art DRRN [4] as our network backbone and change the L2 loss into the proposed NCL loss. More specifically, an enhanced residual unit structure is recursively learned in a recursive block, and several recursive blocks are stacked to learn the residual image between the HR and LR images. The residual image is then added to the input LR image from a global identity branch to estimate the HR image.

Eqn. (11) can be regarded as a smoothed version of Eqn. (10) to improve the generalization ability of the ensemble models. Please note that the optimal value of λ may not necessarily be 0.5 because of the discrepancy between the training and testing data [19]. By setting $\lambda = 0$, we actually achieve conventional ensemble learning (non-boosting type) where each model is optimized independently. It is straightforward to show that the first part in Eqn. (11) corresponds to bias plus an extra term $[K \times \mathbb{V}(G) + \frac{1}{K}\sum_{k=1}^{K}(E[G_k] - E[\tilde{G}])^2]$, while the second part stands for the variance, covariance and the same term $[K \times \mathbb{V}(G) + \frac{1}{K}\sum_{k=1}^{K}(E[G_k] - E[\tilde{G}])^2]$. Since the extra term appears on both sides, it cancels out when we combine them by subtracting, as done in Eqn. (11). Thus by introducing the second part in Eqn. (11), we aim at achieving better "diversity" with negatively correlated base models to balance the components of bias variance, and the ensemble covariance to reduce the overall mean square error (MSE).

To demonstrate this, consider the scenario in Fig. 2. We are using a regression ensemble consisting of 6 regressors where the ground truth is -1.5. Each curve in Fig. 2 illustrates the evolution of one regressor when trained with gradient descent, *i.e.*, $f_{i,n} = f_{i,n-1} - \gamma\frac{d_E}{d_{f_{i,n-1}}}$, where γ and E stands for the learning rate and mean-square loss function, respectively. $i \in \{1, 2, \cdots, 6\}$ is the index of individual models in the ensemble and $n \in \{1, 2, \cdots, 30\}$ stands for the index of iterations. Although both conventional ensemble learning (Fig. 2a) and NCL (Fig. 2b) may lead to correct estimations by simple model averaging, NCL results in much diversified individual models which make error cancellation possible on testing data.

For generalization, consider the artificial spirals dataset in Fig. 1(a), where an ensemble of three single hidden layer feed-forward network (SLFN) is trained on. Then the ensemble is evaluated on data samples densely sampled on $x - y$ plane. The first row in Fig. 1 shows that NCL ensemble leads to more diversified SLFN, compared with conventional ensemble learning as illustrated in the second row of Fig. 1, thus making the resulting ensemble generalize well on testing data. Creating diverse sets of models has been extensively studied, both theoretically [11], [13], [19], [79]–[83] and empirically [72], [84]. More specifically, Breiman [12] derived a VC-type bound for generalization ability of ensemble models which advocated both accurate and decorrelated individual models. In addition, our methods also differ from the classical work of [18] which trains multiple shallow networks.

3.2.3 *Connection with the Rademacher Complexity*

We now show the bound for the Rademacher complexity [85] of the proposed deep negative correlation learning. Firstly we will make no difference between convolution and fully-connected (FC) layers because FC layers can be easily transformed into convolution layers with proper kernel size and padding values.

Definition 1. (Rademacher Complexity). *For a dataset* $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ *generated by a distribution* $\mathcal{D}$ *on set* $\mathcal{X}$ *and a real-valued function class* $\mathcal{G}$ *in* $\mathcal{X}$*, the empirical Rademacher complexity of* G *is the random variable:*

$$\hat{R}_N(\mathcal{G}) = E_\sigma\left[\sup_{G\in\mathcal{G}} \frac{2}{N}\sum_{i=1}^{N}\sigma_i G(\mathbf{x}_i)\right], \tag{13}$$

where $\sigma_1, \cdots, \sigma_N$ *are usually referred to as Rademacher Variable and are independent random variables uniformly chosen from* $\{-1, +1\}$*. The Rademacher complexity of* $\mathcal{G}$ *is* $R_N(\mathcal{G}) = E_X[\hat{R}_N(\mathcal{G})]$.

The empirical Rademacher complexity is widely regarded as the proximity of the generalization ability based on the following theorem:

Theorem 1. (Koltchinskii and Panchenko, 2000 [86]). *Fix* $\delta \in (0, 1)$ *and let* $\mathcal{G}$ *be a class of functions mapping from* X *to [0,1]. Let* $\mathbf{x}_i \in X$ *be drawn independently according to a probability*

distribution $\mathcal{D}$. Then with probability at least $1-\delta$ over random draws of samples of size N, every $G \in \mathcal{G}$ satisfies:

$$E[G(X)] \leq \hat{E}[G(X)] + \hat{R}_N(\mathcal{G}) + 3\sqrt{\frac{\ln(\frac{2}{\delta})}{2N}} \tag{14}$$

where $\hat{E}[G(X)] = \frac{1}{N}\sum_{i=1}^{N} G(\mathbf{x}_i)$.

In addition, we have

Lemma 1. *For $\mathcal{G}$ and $\phi : \mathbb{R} \to \mathbb{R}$, let $\mathcal{G}' := \{\phi \circ G : G \in \mathcal{G}\}$. If ϕ is $\mathcal{L}$-Lipschitz continuous, i.e. $|\phi(x) - \phi(x')| \leq \mathcal{L}|x - x'|$, then for any N:*

$$\hat{R}_N(\mathcal{G}') \leq \mathcal{L}\hat{R}_N(\mathcal{G}) \tag{15}$$

Proof. We provide the proof for $N = 1$, the general case works iteratively.

$$\begin{aligned}
\hat{R}_N(\mathcal{G}') &= E_{\sigma_1}\left[\sup_{G\in\mathcal{G}}(2\sigma_1\phi(G(\mathbf{x}_1)))\right] \\
&= \sup_{G\in\mathcal{G}}(\phi(G(\mathbf{x}_1))) + \sup_{G\in\mathcal{G}}(-\phi(G(\mathbf{x}_1))) \\
&= \sup_{G^1\in\mathcal{G},G^2\in\mathcal{G}}(\phi(G^1(\mathbf{x}_1)) - \phi(G^2(\mathbf{x}_1))) \\
&\leq \sup_{G^1\in\mathcal{G},G^2\in\mathcal{G}}(\mathcal{L}|G^1(\mathbf{x}_1) - G^2(\mathbf{x}_1)|) \\
&= \mathcal{L}(\sup_{G^1\in\mathcal{G}}(G^1(\mathbf{x}_1)) + \sup_{G^2\in\mathcal{G}}(-G^2(\mathbf{x}_1))) \\
&= \mathcal{L}E_{\sigma_1}\left[\sup_{G\in\mathcal{G}}(2\sigma_1 G(\mathbf{x}_1))\right] \\
&= \mathcal{L}\hat{R}_N(\mathcal{G})
\end{aligned} \tag{16}$$

Based on Lemma 1, we have the following conclusion:

Lemma 2. *Let $p \geq 1$, $Z = X \times Y$, $L_p(z) = L_p(\mathbf{x}, \mathbf{y}) = \{\mathbf{x} \to |G(\mathbf{x}) - \mathbf{y}|^p : G \in \mathcal{G}\}$. Assume that $\left[\sup_{G\in\mathcal{G},\mathbf{x}\in X}|G(\mathbf{x}) - \mathbf{y}|\right] \leq M$. Then for any sample X of size N:*

$$\hat{R}_N(L_p) \leq pM^{p-1}\hat{R}_N(G). \tag{17}$$

Proof let $L' = \{\mathbf{x} \to G(\mathbf{x}) - \mathbf{y} : G \in \mathcal{G}\}$. Then we have $L_p = \{\phi \circ l : l \in L'\}$ with $\phi : \mathbf{x} \to |\mathbf{x}|^p$. From Lemma 1, we have ϕ is pM^{p-1} Lipschitz over $[-M, M]$, then we have:

$$\hat{R}_N(L_p) \leq pM^{p-1}\hat{R}_N(L'). \tag{18}$$

Moreover, with

$$\begin{aligned}
\hat{R}_N(L') &= E_\sigma\left[\sup_{G\in\mathcal{G}}(\tfrac{2}{N}\textstyle\sum_{i=1}^{N}(\sigma_i G(\mathbf{x}_i) - \sigma_i y_i))\right] \\
&= E_\sigma\left[\sup_{G\in\mathcal{G}}(\tfrac{2}{N}\textstyle\sum_{i=1}^{N}\sigma_i G(\mathbf{x}_i))\right] + E_\sigma\left[(\tfrac{2}{N}\textstyle\sum_{i=1}^{N}\sigma_i y_i)\right] \\
&= \hat{R}_N(G),
\end{aligned} \tag{19}$$

we complete the proof.

Furthermore, by combining $G(x) = L_2$ as defined in Lemma 2 with Theorem 1, we have

Lemma 3. *Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ be a dataset generated by a distribution $\mathcal{D}$ on set $\mathcal{X}$ and $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$ be their corresponding labels. For a function class $\mathcal{G} \subseteq \{G : X \to \mathbf{Y}\}$ which maps the data X to [0,1], define the commonly used mean square as $\ell_{\mathcal{D}}(G) = E_{\mathcal{D}}[(G(x) - y)]^2$ and the empirical mean square error as as $\hat{\ell}_X(G) = E_X[(G(x) - y)^2]$. Assume that $\left[\sup_{G\in\mathcal{G},\mathbf{x}\in X}|G(x) - \mathbf{y}|\right] \leq M$. Then for a fixed $\delta \in (0, 1)$, with probability at least $1 - \delta$ over random draws of samples of size N, every $G \in \mathcal{G}$ satisfies*

$$\ell_{\mathcal{D}}(G) \leq \hat{\ell}_X(G) + 2M\hat{R}_N(\mathcal{G}) + \sqrt{\frac{ln\frac{2}{\delta}}{2N}}. \tag{20}$$

We now show that the empirical Rademacher complexity of the proposed method on the training set is $\frac{1}{K}$ to the standard network:

Proposition 1. Denote by $\tilde{G} \in \tilde{\mathcal{G}}$ the group convolution based method and by $G \in \mathcal{G}$, the conventional method. Then

$$\hat{R}_N(\tilde{\mathcal{G}}) = \frac{1}{K}\hat{R}_N(\mathcal{G}). \tag{21}$$

Proof.

$$\begin{aligned}
\hat{R}_N(\tilde{\mathcal{G}}) &= E_\sigma\left[\sup_{\tilde{G}\in\tilde{\mathcal{G}}}|\tfrac{2}{N}\textstyle\sum_{i=1}^{N}\sigma_i\tilde{G}(\mathbf{x}_i)|\right] \\
&= E_\sigma\left[\sup_{\tilde{G}\in\tilde{\mathcal{G}}}|\tfrac{2}{N*K}\textstyle\sum_{i=1}^{N}\sigma_i\sum_{k=1}^{K}\tilde{G}_k(\mathbf{x}_i)|\right] \\
&= E_\sigma\left[\sup_{\tilde{G}\in\tilde{\mathcal{G}}}|\tfrac{2}{N*K}\textstyle\sum_{i=1}^{N}\sigma_i\sum_{k=1}^{K}\tilde{G}_k^{\mathcal{Q}-1}(\mathbf{x}_i)\circledast W_k^{\mathcal{Q}}|\right] \\
&= E_\sigma\left[\sup_{G\in\mathcal{G}}|\tfrac{2}{N*K}\textstyle\sum_{i=1}^{N}\sigma_i G(\mathbf{x}_i)\circledast W^{\mathcal{Q}}|\right] \\
&= \frac{1}{K}\hat{R}_N(\mathcal{G}).
\end{aligned} \tag{22}$$

The operation $\sum_{k=1}^{K} G_k^{\mathcal{Q}-1}(\mathbf{x}_i)\circledast W_k$ in Eqn. (22) stands for the convolution operator. $G_k^{\mathcal{Q}-1}(\mathbf{x}_i)$ stand for the k^{th} feature subset of the feature maps $G^{\mathcal{Q}-1}(\mathbf{x}_i)$. More specifically, we divide the feature map of $G^{\mathcal{Q}-1}(\mathbf{x}_i) \subseteq I\!R^{H_i^{\mathcal{Q}-1}\times W_i^{\mathcal{Q}-1}\times C_i^{\mathcal{Q}-1}}$ along the 3^{rd} axis into K subsets. The same procedure is applied to the kernel filter $W^{\mathcal{Q}}$.

Remark 1. *The empirical Rademacher complexity measures the ability of functions from a function class (when applied to a fixed set X) to fit random noise. It is a more modern notion of complexity that is distribution dependent and defined for any class real-valued functions. On the one hand, by setting $K > 1$, our method works in a "divide and conquer" manner and the whole Rademacher complexity is reduced by a factor of K, which, intuitively speaking, makes the function $\tilde{G} \in \tilde{\mathcal{G}}$ easier to learn. On the other hand, K may also affect the term of $\ell_X(G)$. For instance, setting an extremely larger value of K may also lead to a larger value of $\ell_X(G)$ because of much less input feature is provided for each base predictor.*

4 Experiment

In this section, we investigate the feasibility of the proposed method on four regression tasks: crowd counting, personality analysis, age estimation and single image super-resolution. The proposed method is implemented in Caffe [87]. In order to further understand the merits of the proposed methods, we also include some variants of the proposed method. More specifically, for each task, we replace the proposed loss function with L2, SmoothL1 and Tukey loss, and they are referred as "L2", "SmoothL1" and "Tukey", respectively. For the SmoothL1 loss, instead of using a fixed value of 1 for the threshold in Eqn. (1), we treat it as another hyper-parameter and optimized it on the training data. We do not

compare the proposed method explicitly with naive implementations of multiple CNN ensemble as their computational time is much larger and thus are less interested to us. We highlight the best results in each case in **Red**. The second and third best methods are highlighted in Green and Blue, respectively. As different evaluation protocols may be utilized in different applications, we put a ↑ after each metric to indicate the cases wherever a larger value is better. Similarly, ↓ is used in cases wherever smaller value indicates better performance.

4.1 Crowd Counting

For crowd counting, we evaluate the proposed methods on three benchmark datasets: UCF_CC_50 dataset [88], Shanghaitech dataset [33] and WorldExpo'10 dataset [28]. The proposed networks are trained using Stochastic Gradient Descent with a mini-batch size of 1 at a fixed constant momentum value of 0.9. Weight decay with a fixed value of 0.0005 is used as a regularizer. We use a fixed learning rate of 10^{-7} in the last convolution layer of our crowd model to enlarge the gradient signal for effective parameter updating and use a relatively lower learning rate of 10^{-9} in other layers. We set the ensemble size to be 64. More specifically, we use a convolution layer with the kernel of $64 \times 8 \times 1 \times 1$ as regressor $G_k^{\mathcal{Q}}$ on each output feature map to get the final crowd density map. Specifically, each regressor $G_k^{\mathcal{Q}}$ is sparsely connected to a small portion of feature maps from the last convolutional layer ($conv5\_3$) of VGG16 network, implemented via the well-established "group convolution" strategy [30], [89]. We also include another variant of the proposed method called "NCL", which is a shallow network optimized with the same loss function. The details of this network will be elaborated in Section 4.5.

The widely used *mean absolute error* (MAE) and the *root mean squared error* (RMSE) are adopted to evaluate the performance of different methods. The MAE and RMSE are defined as follows:

$$\mathrm{MAE} = \frac{1}{\mathrm{N}} \cdot \sum_{\mathrm{i}=1}^{\mathrm{N}} |(\mathrm{y_i} - \tilde{\mathrm{y}}_\mathrm{i})|, \tag{23}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{N}} \cdot \sum_{\mathrm{i}=1}^{\mathrm{N}} (\mathrm{y_i} - \tilde{\mathrm{y}}_\mathrm{i})^2} \tag{24}$$

Here N represents the total number of images in the testing datasets. The ground truth and the estimated value for the i^{th} image are y_i and $\tilde{y}_i$ respectively.

UCF_CC_50 dataset The challenging UCF_CC_50 dataset [88] contains 50 images that are randomly collected from the Internet. The number of head ranges from 94 to 4543 with an average of 1280 individuals per image. The total number of annotated persons within 50 images is 63974. Challenging issues such as large variations in head number among different images from a small number of training images come in the way of accurately counting for UCF_CC_50 dataset. We follow the standard evaluation protocol by splitting the dataset randomly into five parts in which each part contains ten images. Five-fold cross-validation is employed to evaluate the performance. Since the perspective maps are not provided, we generate the ground truth density map by using the method of Zhang *et al.* [33].

We compare our method on this dataset with the state-of-the-art methods. In [88], [90], [91], handcraft features are used to regress the density map from the input image. Several CNN-based methods in [28], [31], [33]–[35], [92], [93] were also considered here due to their superior performance on this dataset. Table 1 summarizes the detailed results. Firstly, it is obvious that most deep learning methods outperform hand-crafted features significantly. In [31], Boominathan *et al.* proposed to employ a shallow network to assist the training process of deep VGG network. With the proposed deep negative learning strategy, it is also interesting to see that 1) both our deep ("DNCL") and shallow ("NCL") networks work well; 2) deep networks ("DNCL") are better than shallower networks ("NCL"), as expected. However, shallower network still leads to competitive results and may be advantageous in resource-constrained scenarios as it is computationally cheaper; (3) it is straightforward to see that the deeper version of the proposed method outperforms all others on this dataset; (4) the proposed method performs favorably against a naive application of multiple CNN ensemble of [15].

TABLE 1
Comparing results of different methods on the UCF_CC_50 dataset.

Method	Deep Features	MAE ↓	RMSE ↓
Rodriguez *et al.* [90]	✗	655.7	697.8
Lempitsky *et al.* [91]	✗	493.4	487.1
Isrees *et al.* [88]	✗	419.5	541.6
Zhang *et al.* [28]	✓	467.0	498.5
CrowdNet [31]	✓	452.5	-
Zhang *et al.* [33]	✓	377.6	509.1
Zeng *et al.* [92]	✓	363.7	468.4
Mark *et al.* [93]	✓	338.6	424.5
Daniel *et al.* [34]	✓	333.7	425.2
Sam *et al.* [35]	✓	318.1	439.2
Elad *et al.* [15]	✓	364.2	-
L2	✓	394.3	556.9
SmoothL1	✓	384.1	556.7
Tukey	✓	380.7	552.0
NCL	✓	354.1	443.7
DNCL	✓	**288.4**	**404.7**

TABLE 2
Comparison of crowd counting methods on the Shanghaitech dataset.

Method	Part_A		Part_B	
	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓
LBR+RR	303.2	371.0	59.1	81.7
Zhang *et al.* [28]	181.8	277.7	32.0	49.8
Zhang *et al.* [33]	110.2	173.2	26.4	41.3
Sam *et al.* [35]	90.4	135.0	21.6	33.4
Liu *et al.* [94]	-	-	20.8	29.4
L2	105.4	152.3	40.4	58.6
SmoothL1	94.9	150.1	40.3	58.3
Tukey	104.5	151.2	40.3	58.4
NCL	101.7	152.8	25.7	38.6
DNCL	**73.5**	**112.3**	**18.7**	**26.0**

Shanghaitech dataset The Shanghaitech dataset [33] is a large-scale crowd counting dataset, which contains 1198 annotated images with a total of 330,165 persons. This dataset is the largest one in the literature in terms of the number of annotated pedestrians. It consists of two parts: Part_A consisting of 482 images that are randomly captured from the Internet, and Part_B including 716 images that are taken from the busy streets in Shanghai. Each part is divided into training and testing subset. The crowd density varies significantly among the subsets, making it difficult to estimate the number of pedestrians.

We compare our method with six existing methods on the

Shanghaitech dataset. All the detailed results for each method are illustrated in Table 2. In the same way, we can see that all deep learning methods outperform hand-crafted features significantly. The shallow model in [33] employs a much wider structure by a multi-column design and performs better than the shallower CNN models in [28] in both cases. A deeper version of the proposed method performs consistently better than the other shallow one, as expected, because of employing a much deep pre-trained model. Moreover, it is interesting to see that with deep negative learning, even a relatively shallower network structure is on a par with a much complicated and state-of-the-art switching strategy [35]. Finally, our deep structure leads to the best performance in terms of MAE on Part_A and RMSE on Part_B.

WorldExpo'10 dataset The WorldExpo'10 dataset [28] is a large-scale and cross-scene crowd counting dataset. It contains 1132 annotated sequences which are captured by 108 independent cameras, all from Shanghai 2010 WorldExpo'10. This dataset consists of 3980 frames with a total of 199,923 labeled pedestrians, which are annotated at the centers of their heads. Five different regions of interest (ROI) and the perspective maps are provided for the test scenes.

We follow the standard evaluation protocol and use all the training frames to learn our model. For comparison, the quantitative results are given in Table 3. In the same way, we observe that learned representations are more robust than the handcraft features. Even without using the perspective information, our results are still comparable with another deep learning method [28] which used perspective normalization to crop 3×3 square meters patches with 0.5 overlaps on testing time. The deeper version of our proposed method outperforms all other in terms of average performance.

4.2 Personality Analysis

For personality analysis, the ensemble size is set to be 16. We use the ChaLearn personality dataset [3], which consists of $10k$ short video clips with 41.6 hours (4.5M frames) in total. In this dataset, people face and speak to the camera. Each video is annotated with personality attributes as the Big Five personality traits (*Extraversion, Agreeableness, Conscientiousness, Neuroticism and Openness*) [3] in $[0, 1]$. The annotation was done via Amazon Mechanical Turk. For the evaluation, we follow the standard protocol in ECCV 2016 ChaLearn First Impression Challenge [3],

TABLE 3
Comparison of mean absolute error (MAE ↓) of different crowd counting methods on the WorldExpo'10 dataset.

Method	Scenes					Avg.
	S1	S2	S3	S4	S5	
LBP+RR	13.6	58.9	37.1	21.8	23.4	31.0
Zhang *et al.* [28]	9.8	14.1	14.3	22.2	3.7	12.9
Zhang *et al.* [33]	3.4	20.6	12.9	13.0	8.1	11.6
Sam *et al.* [35]	4.4	15.7	10.0	11.0	5.9	9.4
Liu *et al.* [94]	2.0	13.1	**8.9**	17.4	4.8	9.3
L2	3.3	37.9	19.5	10.5	3.7	14.9
SmoothL1	3.7	45.0	30.1	11.1	3.7	18.7
Tukey	3.3	38.3	19.5	10.5	3.7	15.0
NCL	4.9	14.3	18.7	11.3	4.6	10.7
DNCL	**1.9**	**12.1**	20.7	**8.3**	**2.6**	**9.1**

and use the mean accuracy A and coefficient of determination R^2, which are defined as follows:

$$A = 1 - \frac{1}{N^t}\sum_{i}^{N^t}|\mathbf{Y}_i^P - \mathbf{P}_i|, \tag{25}$$

$$R^2 = 1 - \sum_{i}^{N^t}(\mathbf{Y}_i^P - \mathbf{P}_i)^2 / \sum_{i}^{N^t}(\bar{\mathbf{Y}}^P - \mathbf{P}_i)^2, \tag{26}$$

where N^t denotes the total number of testing samples, $\mathbf{Y}^P$ the ground truth, $\mathbf{P}_i$ the prediction, and $\bar{\mathbf{Y}}^P$ the average value of the ground truth.

We train the whole network with an initial learning rate of 0.01. For each mini-batch, we randomly select 10 videos thus generating a total batch size of 100. We set $K = 8$ in this experiment and train the network for $28k$ iterations and decrease the learning rate by a factor of 10 in the $16k^{th}$, $24k^{th}$ and $28k^{th}$ iteration.

The quantitative comparison between the proposed method and other state-of-the-art works on personality recognition is shown in Table 4. Moreover, Table 5 lists the comparison of the details of several latest personality recognition methods. In contrast to other approaches, ours can be trained end-to-end using only one pre-trained model. Moreover, unlike most methods which fuse both acoustic and visual cues, our proposed method uses only video frames as input. The teams from NJU-LAMDA to BU-NKU-v1 are the top five participants in the 1^{st} ChaLearn Challenge on First Impressions [3]. Note that BU-NKU was the only team not using audio in the challenge, and their predictions were rather poor comparatively. After adding the acoustic cues, the same team won the 2^{nd} ChaLearn Challenge on First Impressions [3]. Importantly, our methods only consider visual streams. Firstly, we observe that the deeply learned representations are well transferable between face verification and personality analysis. This can be verified by the last four results in Table 4. By utilizing state-of-the-art face verification network and good practices in video classification [78], those methods outperform current state-of-the-arts. Secondly, L2 and SmoothL1 loss and Tukey Loss all lead to comparably good results for this task. Finally, the proposed method outperforms all the methods on both metrics in all scenarios.

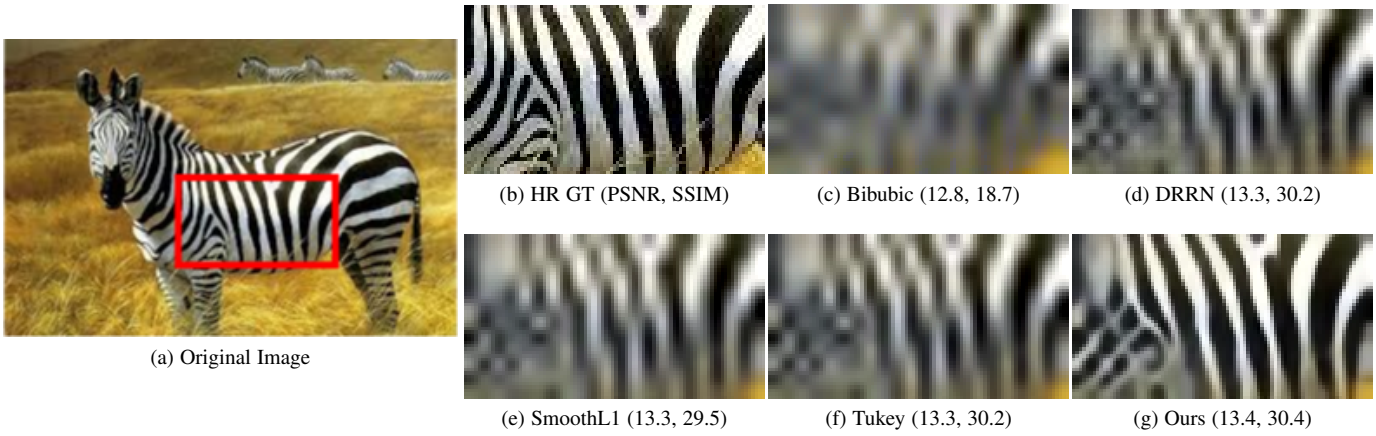
4.3 Age Estimation

We use the same training and evaluation protocol as done in [27]. More specifically, we first use a standard face detector to detect faces [109] and further localized the facial landmarks by AAM [110]. The ensemble size is 5. After that, we perform face alignment to guarantee all eyeballs stay at the same position in the image. We further augment the training data by the following strategies: (1) cropping images with some random offsets, (2) adding Gaussian noise to the original images, and (3) randomly flipping from left to right. We compare the proposed method with various state-of-the-arts on two standard benchmarks: MORPH [95] and FG-NET [96].

As for the evaluation metric, we follow the existing method and choose *Mean Absolute Error* (MAE) as well as *Cumulative Score* (CS). CS is calculated by $CS(l) = \frac{N_l}{N}100\%$, where N is the total number of testing images and N_l is the number of testing facial images whose absolute error between the estimated age and the ground truth age is not greater than l years. Here, we set the same error level 5 as in [27].

TABLE 4
Personality prediction bench-marking using mean accuracy A and coefficient of determination R^2 scores. The results of the first 6 methods are copied from [3] and [43].

	Average		*Extraversion*		*Agreeableness*		*Conscientiousness*		*Neuroticism*		*Openness*	
	$A\uparrow$	$R^2\uparrow$	$A\uparrow$	$R^2\uparrow$	$A\uparrow$	$R^2\uparrow$	$A\uparrow$	$R^2\uparrow$	$A\uparrow$	$R^2\uparrow$	$A\uparrow$	$R^2\uparrow$
NJU-LAMDA	0.913	0.455	0.913	0.481	0.913	0.338	0.917	0.544	0.910	0.475	0.912	0.437
Evolgen	0.912	0.440	0.915	0.515	0.912	0.329	0.912	0.488	0.910	0.455	0.912	0.414
DCC	0.911	0.411	0.911	0.431	0.910	0.296	0.914	0.478	0.909	0.448	0.911	0.402
Ucas	0.910	0.439	0.913	0.489	0.909	0.292	0.911	0.520	0.906	0.457	0.910	0.439
BU-NKU-v1	0.909	0.394	0.916	0.514	0.907	0.234	0.913	0.487	0.902	0.363	0.908	0.372
BU-NKU-v2	0.913	-	0.918	-	0.907	-	0.915	-	0.911	-	0.914	-
L2	0.915	0.467	0.920	0.544	0.912	0.333	0.918	0.543	0.913	0.482	**0.916**	0.426
SmoothL1	0.915	0.466	0.919	0.542	0.912	0.332	0.919	0.548	0.912	0.480	0.913	0.428
Tukey	0.915	0.467	0.919	0.542	0.912	0.332	0.919	0.551	0.912	0.479	0.913	0.430
DNCL	**0.918**	**0.497**	**0.923**	**0.571**	**0.914**	**0.365**	**0.922**	**0.581**	**0.914**	**0.512**	0.915	**0.458**



(a) Original Image (b) HR GT (PSNR, SSIM) (c) Bibubic (12.8, 18.7) (d) DRRN (13.3, 30.2) (e) SmoothL1 (13.3, 29.5) (f) Tukey (13.3, 30.2) (g) Ours (13.4, 30.4)

Fig. 4. Visual comparison for $4\times$ super-resolution of different super-resolution results. Fig. 4b shows the ground-truth high resolution image cropped from the original image in Fig. 4a.

TABLE 5
Comparison of the properties of the proposed method vs. the top teams in the 2016 ChaLearn First Impressions Challenge.

	Fusion	Modality		End-to-End
		Audio	Video	
Ours	late	✗	✓	✓
NJU-LAMDA[1]	late	✓	✓	✓
Evolgen	early	✓	✓	✓
DCC	late	✓	✓	✓
Ucas	late	✓	✓	✗
BU-NKU-v1	early	✗	✓	✗
BU-NKU-v2[2]	early	✓	✓	✗

[1] winner, 1^{st} ChaLearn First Impressions Challenge (ECCV 2016).
[2] winner, 2^{nd} ChaLearn First Impressions Challenge (ICPR 2016)

TABLE 6
Results of different age estimation methods on the MORPH [95] and FG-NET [96] datasets.

Method	MORPH		FG-NET	
	MAE ↓	CS ↑	MAE ↓	CS ↑
Human workers [52]	6.3	51.0	4.70	69.5
AGES [49]	8.83	46.8	6.77	64.1
MTWGP [97]	6.28	52.1	4.83	72.3
CA-SVR [98]	5.88	57.9	4.67	74.5
SVR [99]	5.77	57.1		
LARR [99]			5.07	68.9
OHRank [100]	6.07	56.3	4.48	74.4
DLA [101]	4.77	63.4	4.26	-
Rank [102]	6.49	49.1	5.79	66.5
DIF [52]	-	-	4.80	74.3
CPNN [54]	-	-	4.76	-
CAM [103]	-	-	4.12	-
Rothe et al. [104]	3.45	-	5.01	-
DEX [60]	3.25	-	4.63	-
dLDLF [105]	3.02	81.3	-	-
ARN [58]	3.00	-	-	-
DRF [27]	2.91	82.9	3.85	80.6
SmoothL1	2.99	82.5	3.95	80.9
Tukey	2.90	83.1	3.87	**82.5**
DNCL	**2.85**	**83.8**	**3.71**	81.8

We first summarize our results on MPRPH dataset in Table 6. It contains more than 55,000 images from about 13,000 people of different races. We perform our evaluation on the first setting (setting I) [27], which selects 5,492 images of Caucasian Descent people from the original MORPH dataset to reduce the cross-ethnicity effects. In this setting, these 5,492 images are randomly partitioned into two subsets: 80% of the images are selected for training and others for testing. The random partition is repeated 5 times, and the final performance is averaged over these 5 different partitions. Since the DRF method [27] assumed each leaf node was a normal distribution, minimizing negative log-likelihood loss was equivalent to minimizing the L2 loss of each node [1]. As can be seen from Table 6, the proposed method achieves the best

1. Actually the released implementation of DRF [27] used L2 loss to avoid observing negative loss during training.

TABLE 7
Average PSNR/SSIM/IFC score for image super-resolution of scale factor $\times 2$, $\times 3$ and $\times 4$ on datasets Set5, Set14, BSD100 and Urban100.

Dataset		PSNR/SSIM/IFC ↑								
		SRCNN [61]	SelfEx [106]	RFL [107]	VDSR [108]	DSRN [67]	DRRN [4]	Tukey	SmoothL1	DNCL
set5	×2	36.7/95.4/8.04	36.5/95.4/7.81	36.5/95.4/8.56	37.5/95.9/8.57	37.7/95.9/8.6	37.7/95.9/8.67	37.5/**95.9**/7.58	37.0/95.6/7.08	**37.8/95.9/8.70**
	×3	32.8/90.9/4.66	32.6/90.9/4.75	32.4/90.6/4.93	33.7/92.1/5.22	33.9/92.2/5.22	34.0/92.4/5.49	33.1/91.6/4.65	33.0/91.6/4.58	**34.1/92.5/5.43**
	×4	30.5/86.3/2.99	30.3/86.2/3.17	30.1/85.5/3.19	31.6/88.4/3.55	31.4/88.3/3.50	31.7/88.9/3.70	30.8/87.4/2.78	30.4/87.1/2.73	**31.7/89.0/3.73**
set14	×2	32.5/90.7/7.78	32.2/90.3/7.59	32.3/90.4/8.18	33.0/91.2/8.18	33.2/91.3/8.17	**33.2**/91.4/**8.32**	32.7/86.9/7.48	32.2/87.0/7.10	**33.2/91.4**/8.30
	×3	29.3/82.2/4.34	29.2/82.0/4.37	29.1/81.6/4.53	29.8/83.1/4.73	30.3/83.7/**4.89**	30.0/83.5/4.88	29.4/82.7/4.22	29.5/82.4/4.17	**30.0/83.5**/4.89
	×4	27.5/75.1/2.75	27.4/75.2/2.89	27.2/74.5/2.92	28.0/76.7/3.13	28.1/77.0/3.15	28.2/77.2/3.25	27.7/76.0/2.87	27.1/75.4/2.84	**28.3/77.3/3.28**
BSD100	×2	31.4/0.89/-	31.2/88.6/-	31.2/88.4/-	31.9/89.6/-	**32.1**/89.7/-	32.1/89.7/-	31.7/88.4/-	31.6/88.3/-	32.1/**89.8**/-
	×3	28.4/78.6/-	28.3/78.4/-	28.2/78.1/-	28.8/79.8/-	28.8/79.7/-	29.0/80.0/-	28.6/79.3/-	28.2/78.8/-	**29.0/80.1**/-
	×4	26.9/71.0/-	26.8/71.1/-	26.8/70.5/-	27.3/72.5/-	27.3/72.4/-	27.4/72.8/-	27.0/71.8/-	27.0/70.8/-	**27.4/72.9**/-
Urban100	×2	29.4/89.5/7.99	29.5/89.7/7.94	29.1/89.0/8.45	30.8/91.4/8.65	31.0/91.6/8.60	31.2/91.9/8.92	30.3/90.9/7.87	30.1/90.1/7.25	**31.3/92.0/8.95**
	×3	26.2/79.9/4.58	26.4/80.9/4.84	25.9/79.0/4.80	27.1/82.8/5.19	27.2/82.8/5.17	27.5/83.8/**5.46**	26.6/81.4/4.58	26.5/81.4/4.52	**27.6/83.8**/5.47
	×4	24.5/72.2/2.96	24.8/73.7/3.31	24.2/71.0/3.11	25.2/75.2/3.50	25.1/74.7/3.30	25.4/76.4/3.68	24.7/73.3/2.93	24.1/73.2/2.86	**25.6/76.5/3.71**

performance on this dataset and outperforms the current state-of-the-arts with a clear margin.

We then conduct experiments on FG-NET [96], which contains 1002 facial images of 82 individuals. Each individual in FG-NET has more than 10 photos taken at different ages. The FG-NEt data is challenging because each image may have a large variation in lighting conditions, poses, and expressions. We follow the protocol of [27] to perform "leave one out" cross-validation on this dataset. The quantitative comparisons on FG-NET dataset are shown in Table 6. As can be seen, our method achieves better results (MSE: 3.71 vs 3.85 and CS: 81.8 vs 80.6) than DRF [27].

4.4 Image Super-resolution

We follow exactly the same training and evaluation protocol. More specifically, by following [107], [108], a training dataset of 291 images, where 91 images are from Yang et al. [111] and other 200 images are from Berkeley Segmentation Dataset [112], were utilized for training. Finally, the method was evaluated on Set5 [113], Set14 [114], BSD100 [112] and Urban100 [106] dataset, which have 5, 14, 100 and 100 images respectively. The initial learning rate is set to 0.1 and then decreased to half every 10 epochs. Since a large learning rate is used in our work, we adopt the adjustable gradient clipping [4] to boost the convergence rate while suppressing exploding gradients. Peak Signal-to-Noise Ratio (PSNR), Structural SIMilarity (SSIM) [115] and Information Fidelity Criterion (IFC) [116] were used for the quantitative evaluations.

Table 7 summarizes the main results of both PSNR in db and SSIM ($\times 100\%$) on the four testing sets. Similarly, the results of IFC are presented in Table 7. Firstly, we can find that the image super-resolution is extremely challenging as most state-of-the-art approaches perform comparably well. However, it is still obvious that the proposed method outperforms the original L2 loss in most cases, leading to even better results than a more recent work using dual-state recurrent networks [67]. In addition, other loss functions such as SmoothL1 and Tukey loss are both outperformed by L2 loss in a large margin. Qualitative comparisons among DRRN [4] and SmoothL1, Tukey, and our proposed method are illustrated in Fig. 4. As we can see, our method produces relatively sharper edges with respect to patterns, while other methods may give blurry results.

4.5 Discussions

After demonstrating the superiority of the proposed method by extensively comparing them with many state-of-the-art methods on multiple datasets, we now provide more discussions to shed light upon their rationale and sensitivities with some hyper-parameters.

TABLE 8
Comparing the performance of NCL and conventional ensemble on the crowd counting task.

Datasets	Conventional Ensemble		DNCL	
	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓
UCF_CC_50	380.5	527.2	**288.4**	**404.7**
Shanghaitech Part_A	91.6	127.9	**73.5**	**112.3**
Shanghaitech Part_B	21.3	30.9	**18.7**	**26.0**
WorldExpo'10	16.4	-	**9.1**	-

NCL or Conventional Ensemble Learning? In Table 8, we compared the performance of the proposed method with conventional ensemble learning and choose crowd counting as a study case. It is widely accepted that training deep networks like VGG remains to be challenging. In [31], a shallow network was proposed to assist the training and improve the performance of deep VGG network. When compared with results achieved on dataset UCF_CC_50 by other methods shown in Table 1, our implementation of a conventional ensemble method using a single VGG network leads to clearly improved results. However, it still over-fits severely compared with other state-of-the-art methods. More specifically, it was outperformed by recent methods such as multi-column structure [33], multi-scale Hydra method [34], and advanced switching strategy [35]. In contrast, the proposed method leads to much-improved performance compared with this baseline in all cases and outperforms all the aforementioned methods. As illustrated in Fig. 2, the NCL mechanism used here encourages diversities in the ensemble, and thus it is more likely to allow error cancelling. For more results on other datasets, please refer to Table 9 and Table 10.

The learning objective function in Eqn. (11) is also in line with Breiman's strength-correlation theory [12] on the VC-type bound for the generalization ability of ensemble models, which advocated both accurate and decorrelated individual models. It as well appreciated that the individual model should be able to exhibit different patterns of generalization– a very simple intuitive explanation is that a million identical estimators are obviously no better than a single.

TABLE 9
Results of different ensemble strategies on the age and crowd datasets.

Dataset	MORPH		ShanghaiA	
Metric	MAE↓	CS↑	MAE↓	RMSE↓
Proposed	**2.35**	**83.8**	**73.5**	**112.3**
L1-NCL	2.94	82.8	77.9	118.86
Tukey-NCL	2.87	83.4	86.3	121.7
Conventional Ensemble	2.89	83.1	91.6	127.9

TABLE 10
Results of different ensemble strategies on the personality and the Urban100 (scale factor of $\times 4$) dataset.

	Chalearn (average score)		Urban100 (4×4)
	$A \uparrow$	$R^2 \uparrow$	PSNR/SSIM/IFC ↑
Proposed	**0.918**	**0.497**	**25.6/76.5/3.71**
L1-NCL	0.916	0.468	24.6/75.2/3.46
Tukey-NCL	0.915	0.467	24.9/74.1/3.21
Conventional Ensemble	0.917	0.470	24.7/75.0/3.44

DNCL for other loss functions. It is widely-accepted that encouraging diversity could generate a better ensemble. Although DNCL is derived under the commonly used L2 loss function, here we show that naively applying this idea to other loss functions can be beneficial. To this end, we replace the first part in Eqn. (11) with other loss functions while keeping the second part unchanged to make the ensemble negatively-correlated. We report the detailed results in Table 9 and Table 10. Firstly, the results show that the proposed ensemble strategy still generates better results than single model for each loss function but is outperformed by the proposed method. Secondly, one can observe that in some cases NCL with other loss functions was outperformed by a conventional ensemble. This indicates that other diversity measurements [117] could be better when other loss functions were utilized.

Effect of λ and K. Parameter λ controls the correlation between each model in the ensemble. On the one hand, setting $\lambda = 0$ is equivalent to train each regressor in an independent manner. On the other hand, employing a larger value for λ overemphasizes the effect of diversity and may lead to poor individual regressors. We empirically find that setting λ to be a relatively smaller value $\lambda \in [10^{-3}, 10^{-2}]$ usually leads to satisfactory results. Parameter K stands for the number of base regressors in the ensemble. Theoretically speaking, conventional ensemble learning such as bagging and decision tree ensemble requires larger ensemble sizes [72], [73], [118] to perform well. However, with the constraint of using the same amount of parameter, increasing the value of K will pass each base model less input information, which may lead to worse performance. We empirically find that the proposed method works well even with a relatively smaller ensemble size. For crowd counting, setting K to be within 32 and 64 can generate satisfactory results, and it is set to be 64 by default as no significant improvement is observed with a more number of regressors. A more detailed report on the effect of K is provided in Table 11. Similarly, the performances of personality analysis and image super-resolution are stable when K is within [8,16] and [16,32], and they are set to be 8 and 16, respectively. For age estimation, we use the same ensemble size of 5 as done in the original paper [27].

TABLE 11
Effect of K on Shanghai Part A dataset.

K	1	16	32	64	128	256	512
MAE↓	105.4	94.1	79.1	**73.5**	179.4	433.3	433.5
RMSE↓	152.3	138.8	113.1	**112.3**	257.1	560.2	560.2

Independent of the network backbone. While tremendous progress has been achieved in vision community by aggressively exploring deeper [119] or wider architectures [120], specially-designed network architecture [121]–[123], or heuristic engineering tricks [124] with the standard "convolution + pooling" recipe, we want to emphasize that the proposed method is independent of the network backbone and almost complementary to those strategies. To show this, we first observe that combing the proposed NCL learning strategies with each "special-purpose" network in each task can lead to improved results. In order to further demonstrate the independence between the proposed method and the network backbone, we choose crowd counting as an example and train a relatively shallower model named as NCL, which is constructed by stacking several Multi-Scale Blob as shown in Fig. 5, aiming to increase the depth and expand the width of the crowd model in a single network. Multi-Scale Blob (MSB) is an Inception-like model which enhances the feature diversity by combining feature maps from different network branches. More specifically, it contains multiple filters with different kernel size (including 7×7, 5×5 and 3×3). This also makes the net more sensitive to crowd scale changing of the images.

Motivated by VGGNet [125], to make the model more discriminative, we further achieve 5×5 and 7×7 convolutional layers by stacking two and three 3×3 convolutional layers, respectively. In our adopted network, the first convolution layer consists of $16 \times 5 \times 5$ filters and is followed by a 2×2 max pooling layer. After that, we stack two MSB modules, as demonstrated in Fig. 5, where the first MSB module is followed by a 2×2 max-pooling layer. The number of feature maps of each convolution layer in these two MSB modules is 24 and 32, respectively. Finally, we use the same 1×1 convolution layer on each of the feature maps as regressor $G_k^{\mathcal{Q}}$ to get the final crowd density map.

The main results of the shallow network can be found in Table 1, Table 2, and Table 3. With the proposed negative correlation learning strategy, it is also interesting to see that 1)both our deep and shallow networks work well; 2) deep networks (DNCL) are better than shallower networks (NCL), as expected. However, the shallower network (NCL) still leads to competitive results and may be advantageous in resource-constrained scenarios as it is computationally cheaper.

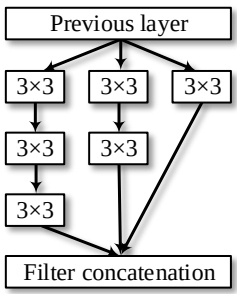


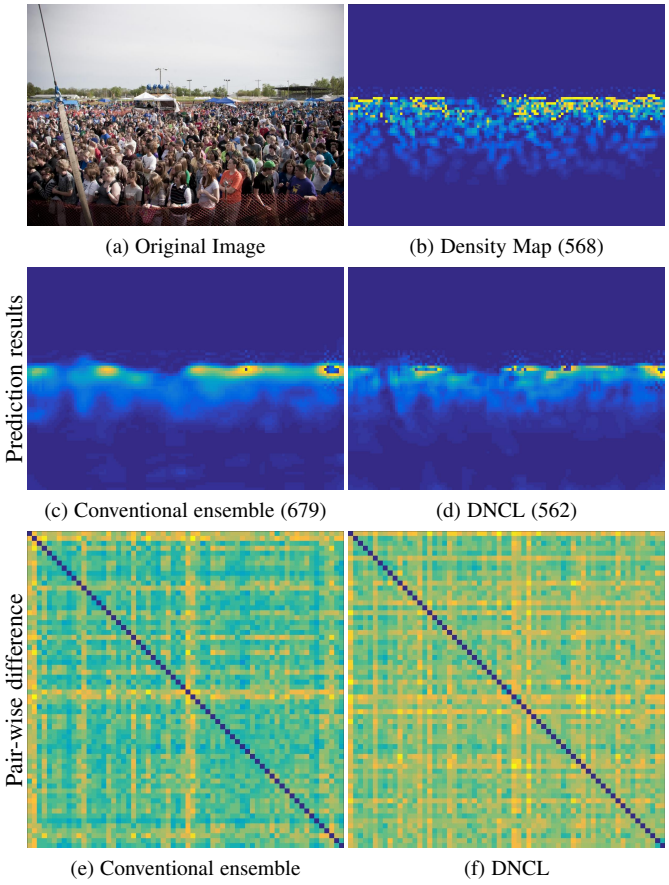
Fig. 5. The Multi-Scale Blob module used in NCL.

Other Aggregation Methods Our loss function is derived under the widely-used ensemble setting in which each base model is assigned with equal importance. In this part, we investigate the effect of DNCL when different base models have different importance. To this end, we use another 1×1 convolution to aggregate the results from each base model and report the results the on Shanghaitech Part A dataset, and the results are summarized in Table 12. The experimental results show that the proposed methods achieve better results. However, as the diversities are also enhanced in the "weighted average" method, the results are better than conventional ensemble, as expected.

TABLE 12
Comparison of different aggregation methods on the Shanghaitech Part A dataset.

	MAE	RMSE
Conventional Ensemble	91.6	127.9
DNCL	**73.5**	**112.3**
Weighted average	83.5	120.8

Visualization of the Enhanced Diversities. In this section, we provide other evidence to show the enhanced diversities in our ensemble methods. We choose crowd counting as our studying case and compute their pair-wise Euclidean distance between each pair of the predictions from each base model. From Fig. 6e and Fig. 6f, we can easily observe that there exist a larger discrepancy in the proposed method, as we expected. Finally, a more diversified ensemble leads to better final performance, as can be found in Fig. 6b, 6c and 6d.



(a) Original Image (b) Density Map (568)
(c) Conventional ensemble (679) (d) DNCL (562)
(e) Conventional ensemble (f) DNCL

Fig. 6. Visualization on the diversities with all 64 base models. The first row shows the input image and the ground-truth number of people. The second row shows the predicted density map from conventional ensemble and NCL, respectively. The number in the bracket represents the ground-truth number of people. The third row shows the pair-wise Euclidean distance between the predictions of individual base models in conventional ensemble and NCL, respectively. It can be seen that the proposed method leads to much diversified base models which can yield better overall performances.

5 Conclusion

In this paper, we present a simple yet effective learning strategy for regression. We pose a typical deep regression network as an ensemble learning problem and learn a pool of weak regressors using convolutional feature maps. The main component of this ensemble architecture is the introduction of negative correlation learning (NCL), which aims to improve the generalization capability of the ensemble models. We show the proposed method has sound generalization capability through managing their intrinsic diversities. The proposed method is generic and independent of the backbone network architectures. Extensive experiments on several challenging tasks, including crowd counting, personality analysis, age estimation, and image super-resolution demonstrate the superiority of the proposed method over other loss functions and current state-of-the-art.

Acknowledgments

This research was supported by NSFC (61922046, 61620106008), the national youth talent support program, and the Tianjin Natural Science Foundation (18ZXZNGX00110, 17JCJQJC43700).

References

[1] Z. Shi, L. Zhang, Y. Liu, X. Cao, Y. Ye, M.-M. Cheng, and G. Zheng, "Crowd counting with deep negative correlation learning," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 5382–5390.

[2] Z. Huo, X. Yang, C. Xing, Y. Zhou, P. Hou, J. Lv, and X. Geng, "Deep age distribution learning for apparent age estimation," in *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2016, pp. 17–24.

[3] V. Ponce-López, B. Chen, M. Oliu, C. Corneanu, A. Clapés, I. Guyon, X. Baró, H. J. Escalante, and S. Escalera, "Chalearn LAP 2016: First round challenge on first impressions – dataset and results," in *Eur. Conf. Comput. Vis.*, 2016.

[4] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *IEEE Conf. Comput. Vis. Pattern Recog.*, vol. 1, no. 4, 2017.

[5] L. Zhang, J. Varadarajan, P. N. Suganthan, N. Ahuja, and P. Moulin, "Robust visual tracking using oblique random forests," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[6] R. Girshick, "Fast R-CNN," in *Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.

[7] P. J. Huber *et al.*, "Robust estimation of a location parameter," *The annals of mathematical statistics*, vol. 35, no. 1, pp. 73–101, 1964.

[8] V. Belagiannis, C. Rupprecht, G. Carneiro, and N. Navab, "Robust optimization for deep regression," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 2830–2838.

[9] P. J. Huber, "Robust statistics," in *International Encyclopedia of Statistical Science*. Springer, 2011, pp. 1248–1251.

[10] M. J. Black and A. Rangarajan, "On the unification of line processes, outlier rejection, and robust statistics with applications in early vision," *Int. J. Comput. Vis.*, vol. 19, no. 1, pp. 57–91, 1996.

[11] T. G. Dietterich *et al.*, "Ensemble methods in machine learning," *Multiple Classifier Systems*, vol. 1857, pp. 1–15, 2000.

[12] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[13] Y. Ren, L. Zhang, and P. N. Suganthan, "Ensemble classification and

regression-recent developments, applications and future directions," *IEEE Comput. Intell. Mag.*, vol. 11, no. 1, pp. 41–53, 2016.

[14] X. Qiu, L. Zhang, P. N. Suganthan, and G. A. Amaratunga, "Oblique random forest ensemble via least square estimation for time series forecasting," *Information Sciences*, vol. 420, pp. 249–262, 2017.

[15] E. Walach and L. Wolf, "Learning to count with cnn boosting," in *Eur. Conf. Comput. Vis.* Springer, 2016, pp. 660–676.

[16] S. Han, Z. Meng, A.-S. Khan, and Y. Tong, "Incremental boosting convolutional neural network for facial action unit recognition," in *Conf. Neural Inf. Process. Syst.*, 2016, pp. 109–117.

[17] C. Cortes, M. Mohri, and U. Syed, "Deep boosting," in *Int. Conf. Mach. Learn.*, 2014, pp. 1179–1187.

[18] Y. Liu, X. Yao, and T. Higuchi, "Evolutionary ensembles with negative correlation learning," *IEEE Trans. Evol. Comput.*, vol. 4, no. 4, pp. 380–387, 2000.

[19] G. Brown, J. L. Wyatt, and P. Tiňo, "Managing diversity in regression ensembles," *JMLR*, vol. 6, no. Sep, pp. 1621–1650, 2005.

[20] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Conf. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[21] Y. Sun, X. Wang, and X. Tang, "Deep convolutional network cascade for facial point detection," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 3476–3483.

[22] A. Toshev and C. Szegedy, "Deeppose: Human pose estimation via deep neural networks," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 1653–1660.

[23] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Facial landmark detection by deep multi-task learning," in *Eur. Conf. Comput. Vis.* Springer, 2014, pp. 94–108.

[24] X. Wang, L. Zhang, L. Lin, Z. Liang, and W. Zuo, "Deep joint task learning for generic object extraction," in *Conf. Neural Inf. Process. Syst.*, 2014, pp. 523–531.

[25] I. Barandiaran, "The random subspace method for constructing decision forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, 1998.

[26] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[27] W. Shen, Y. Guo, Y. Wang, K. Zhao, B. Wang, and A. L. Yuille, "Deep regression forests for age estimation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, June 2018.

[28] C. Zhang, H. Li, X. Wang, and X. Yang, "Cross-scene crowd counting via deep convolutional neural networks," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 833–841.

[29] C. Wang, H. Zhang, L. Yang, S. Liu, and X. Cao, "Deep people counting in extremely dense crowds," in *ACM Int. Conf. Multimedia*. ACM, 2015, pp. 1299–1302.

[30] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Conf. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[31] L. Boominathan, S. S. Kruthiventi, and R. V. Babu, "Crowdnet: a deep convolutional network for dense crowd counting," in *ACM Int. Conf. Multimedia*. ACM, 2016, pp. 640–644.

[32] Z. Shi, L. Zhang, Y. Sun, and Y. Ye, "Multiscale multitask deep netvlad for crowd counting," *IEEE Trans Ind. Informat.*, vol. 14, no. 11, pp. 4953–4962, 2018.

[33] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image crowd counting via multi-column convolutional neural network," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 589–597.

[34] D. Onoro-Rubio and R. J. López-Sastre, "Towards perspective-free object counting with deep learning," in *Eur. Conf. Comput. Vis.* Springer, 2016, pp. 615–629.

[35] D. B. Sam, S. Surya, and R. V. Babu, "Switching convolutional neural network for crowd counting," in *IEEE Conf. Comput. Vis. Pattern Recog.*, vol. 1, no. 3, 2017, p. 6.

[36] C. Arteta, V. Lempitsky, and A. Zisserman, "Counting in the wild," in *ECCV*. Springer, 2016, pp. 483–498.

[37] M. Noroozi, H. Pirsiavash, and P. Favaro, "Representation learning by learning to count," in *ICCV*, 2017, pp. 5898–5906.

[38] B. Lepri, R. Subramanian, K. Kalimeri, J. Staiano, F. Pianesi, and N. Sebe, "Connecting meeting behavior with extraversion — a systematic study," *IEEE Trans Aff. Comput.*, vol. 3, no. 4, pp. 443–455, 2012.

[39] J.-I. Biel, L. Teijeiro-Mosquera, and D. Gatica-Perez, "Facetube: predicting personality from facial expressions of emotion in online conversational video," in *ACM Int. Conf. Multimodal Interaction*. ACM, 2012, pp. 53–56.

[40] D. Sanchez-Cortes, J.-I. Biel, S. Kumano, J. Yamato, K. Otsuka, and D. Gatica-Perez, "Inferring mood in ubiquitous conversational video," in *International Conference on Mobile and Ubiquitous Multimedia*. ACM, 2013, p. 22.

[41] M. K. Abadi, J. A. M. Correa, J. Wache, H. Yang, I. Patras, and N. Sebe, "Inference of personality traits and affect schedule by analysis of spontaneous reactions to affective videos," in *IEEE Conf. Autom. Face. Gest. Recog.*, vol. 1. IEEE, 2015, pp. 1–8.

[42] L. M. Batrinca, N. Mana, B. Lepri, F. Pianesi, and N. Sebe, "Please, tell me about yourself: automatic personality assessment using short self-presentations," in *ACM Int. Conf. Multimodal Interaction*. ACM, 2011, pp. 255–262.

[43] C.-L. Zhang, H. Zhang, X.-S. Wei, and J. Wu, "Deep bimodal regression for apparent personality analysis," in *Eur. Conf. Comput. Vis.* Springer, 2016, pp. 311–324.

[44] Y. Güçlütürk, U. Güçlü, M. A. van Gerven, and R. van Lier, "Deep impression: Audiovisual deep residual networks for multimodal apparent personality trait recognition," in *Eur. Conf. Comput. Vis.* Springer, 2016, pp. 349–358.

[45] A. Subramaniam, V. Patel, A. Mishra, P. Balasubramanian, and A. Mittal, "Bi-modal first impressions recognition using temporally ordered deep audio and stochastic visual features," in *Eur. Conf. Comput. Vis.* Springer, 2016, pp. 337–348.

[46] F. Gürpınar, H. Kaya, and A. A. Salah, "Combining deep facial and ambient features for first impression estimation," in *Eur. Conf. Comput. Vis.* Springer, 2016, pp. 372–385.

[47] J. Junior, C. Jacques, Y. Güçlütürk, M. Pérez, U. Güçlü, C. Andujar, X. Baró, H. J. Escalante, I. Guyon, M. A. van Gerven *et al.*, "First impressions: A survey on computer vision-based apparent personality trait analysis," *arXiv preprint arXiv:1804.08046*, 2018.

[48] H. J. Escalante, H. Kaya, A. A. Salah, S. Escalera, Y. Gucluturk, U. Guclu, X. Baro, I. Guyon, J. J. Junior, M. Madadi *et al.*, "Explaining first impressions: Modeling, recognizing, and explaining apparent personality from videos," *arXiv preprint arXiv:1802.00745*, 2018.

[49] X. Geng, Z.-H. Zhou, and K. Smith-Miles, "Automatic age estimation based on facial aging patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2234–2240, 2007.

[50] G. Guo, G. Mu, Y. Fu, and T. S. Huang, "Human age estimation using bio-inspired features," in *IEEE Conf. Comput. Vis. Pattern Recog.* IEEE, 2009, pp. 112–119.

[51] G. Guo and G. Mu, "Simultaneous dimensionality reduction and human age estimation via kernel partial least squares regression," in *IEEE Conf. Comput. Vis. Pattern Recog.* IEEE, 2011, pp. 657–664.

[52] H. Han, C. Otto, X. Liu, and A. K. Jain, "Demographic estimation from face images: Human vs. machine performance," *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 6, pp. 1148–1161, 2015.

[53] A. Montillo and H. Ling, "Age regression from faces using random forests," in *IEEE Int. Conf. Image Process.* IEEE, 2009, pp. 2465–2468.

[54] X. Geng, C. Yin, and Z.-H. Zhou, "Facial age estimation by learning from label distributions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 10, pp. 2401–2412, 2013.

[55] D. Yi, Z. Lei, and S. Z. Li, "Age estimation by multi-scale convolutional network," in *Asian Conf. Comput. Vis.* Springer, 2014, pp. 144–158.

[56] Z. Niu, M. Zhou, L. Wang, X. Gao, and G. Hua, "Ordinal regression with multiple output cnn for age estimation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 4920–4928.

[57] S. Chen, C. Zhang, M. Dong, J. Le, and M. Rao, "Using ranking-cnn for age estimation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[58] E. Agustsson, R. Timofte, and L. Van Gool, "Anchored regression networks applied to age estimation and super resolution," in *Int. Conf. Comput. Vis.* IEEE, 2017, pp. 1652–1661.

[59] K. Li, J. Xing, C. Su, W. Hu, Y. Zhang, and S. Maybank, "Deep cost-sensitive and order-preserving feature learning for cross-population age estimation," in *Int. Conf. Comput. Vis.*, 2018.

[60] R. Rothe, R. Timofte, and L. Van Gool, "Deep expectation of real and apparent age from a single image without facial landmarks," *Int. J. Comput. Vis.*, vol. 126, no. 2-4, pp. 144–157, 2018.

[61] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, 2016.

[62] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 1874–1883.

[63] Y. Zhang, Y. Zhang, J. Zhang, D. Xu, Y. Fu, Y. Wang, X. Ji, and Q. Dai, "Collaborative representation cascade for single-image super-resolution," *IEEE Trans. Syst. Man Cy.-S.*, no. 99, pp. 1–16, 2017.

[64] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 2472–2481.

[65] J. Kim, J. Kwon Lee, and K. Mu Lee, "Deeply-recursive convolutional network for image super-resolution," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 1637–1645.

[66] X. Mao, C. Shen, and Y.-B. Yang, "Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections," in *Conf. Neural Inf. Process. Syst.*, 2016, pp. 2802–2810.

[67] W. Han, S. Chang, D. Liu, M. Yu, M. Witbrock, and T. S. Huang, "Image super-resolution via dual-state recurrent networks," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

[68] M. Haris, G. Shakhnarovich, and N. Ukita, "Deep backprojection networks for super-resolution," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

[69] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *ECCV*, 2018, pp. 286–301.

[70] Y. Zhang, K. Li, K. Li, B. Zhong, and Y. Fu, "Residual non-local attention networks for image restoration," in *ICLR*, 2019.

[71] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, and L. Zhang, "Ntire 2017 challenge on single image super-resolution: Methods and results," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017, pp. 114–125.

[72] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems," *JMLR*, vol. 15, no. 1, pp. 3133–3181, 2014.

[73] L. Zhang and P. N. Suganthan, "Benchmarking ensemble classifiers with novel co-trained kernal ridge regression and random vector functional link ensembles [research frontier]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 61–72, 2017.

[74] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.

[75] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.

[76] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," in *IEEE Conf. Comput. Vis. Pattern Recog.*, vol. 1, 2017, p. 1.

[77] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[78] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *Eur. Conf. Comput. Vis.* Springer, 2016, pp. 20–36.

[79] G. Brown, J. Wyatt, R. Harris, and X. Yao, "Diversity creation methods: a survey and categorisation," *Information Fusion*, vol. 6, no. 1, pp. 5–20, 2005.

[80] L. L. Minku, A. P. White, and X. Yao, "The impact of diversity on online ensemble learning in the presence of concept drift," *IEEE TKDE*, vol. 22, no. 5, pp. 730–742, 2009.

[81] S. Lee, S. P. S. Prakash, M. Cogswell, V. Ranjan, D. Crandall, and D. Batra, "Stochastic multiple choice learning for training diverse deep ensembles," in *NeurIPS*, 2016, pp. 2119–2127.

[82] M. Alhamdoosh and D. Wang, "Fast decorrelated neural network ensembles with random weights," *Information Sciences*, vol. 264, pp. 104–117, 2014.

[83] Z.-H. Zhou, J. Wu, and W. Tang, "Ensembling neural networks: many could be better than all," *Artificial Intelligence*, vol. 137, no. 1-2, pp. 239–263, 2002.

[84] L. K. Hansen and P. Salamon, "Neural network ensembles," *IEEE TPAMI*, no. 10, pp. 993–1001, 1990.

[85] P. L. Bartlett and S. Mendelson, "Rademacher and gaussian complexities: Risk bounds and structural results," *JMLR*, vol. 3, no. Nov, pp. 463–482, 2002.

[86] V. Koltchinskii, D. Panchenko *et al.*, "Empirical margin distributions and bounding the generalization error of combined classifiers," *The Annals of Statistics*, vol. 30, no. 1, pp. 1–50, 2002.

[87] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *CoRR*, vol. abs/1408.5093, 2014.

[88] H. Idrees, I. Saleemi, C. Seibert, and M. Shah, "Multi-source multi-scale counting in extremely dense crowd images," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 2547–2554.

[89] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Stct: Sequentially training convolutional networks for visual tracking," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 1373–1381.

[90] M. Rodriguez, I. Laptev, J. Sivic, and J.-Y. Audibert, "Density-aware person detection and tracking in crowds," in *IEEE Conf. Comput. Vis. Pattern Recog.* IEEE, 2011, pp. 2423–2430.

[91] V. Lempitsky and A. Zisserman, "Learning to count objects in images," in *Conf. Neural Inf. Process. Syst.*, 2010, pp. 1324–1332.

[92] L. Zeng, X. Xu, B. Cai, S. Qiu, and T. Zhang, "Multi-scale convolutional neural networks for crowd counting," *arXiv preprint arXiv:1702.02359*, 2017.

[93] M. Marsden, K. McGuiness, S. Little, and N. E. O'Connor, "Fully convolutional crowd counting on highly congested scenes," *arXiv preprint arXiv:1612.00220*, 2016.

[94] J. Liu, C. Gao, D. Meng, and A. G. Hauptmann, "Decidenet: counting varying density crowds through attention guided detection and density estimation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018, pp. 5197–5206.

[95] K. Ricanek and T. Tesafaye, "Morph: A longitudinal image database of normal adult age-progression," in *IEEE Conf. Autom. Face. Gest. Recog.* IEEE, 2006, pp. 341–345.

[96] G. Panis, A. Lanitis, N. Tsapatsoulis, and T. F. Cootes, "Overview of research on facial ageing using the fg-net ageing database," *IET Biometrics*, vol. 5, no. 2, pp. 37–46, 2016.

[97] Y. Zhang and D.-Y. Yeung, "Multi-task warped gaussian process for personalized age estimation," in *IEEE Conf. Comput. Vis. Pattern Recog.* IEEE, 2010, pp. 2622–2629.

[98] K. Chen, S. Gong, T. Xiang, and C. Change Loy, "Cumulative attribute space for age and crowd density estimation," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 2467–2474.

[99] G. Guo, Y. Fu, C. R. Dyer, and T. S. Huang, "Image-based human age estimation by manifold learning and locally adjusted robust regression," *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1178–1188, 2008.

[100] K.-Y. Chang, C.-S. Chen, and Y.-P. Hung, "Ordinal hyperplanes ranker with cost sensitivities for age estimation," in *IEEE Conf. Comput. Vis. Pattern Recog.* IEEE, 2011, pp. 585–592.

[101] X. Wang, R. Guo, and C. Kambhamettu, "Deeply-learned feature for age estimation," in *IEEE Winter Conference on Computer Vision*. IEEE, 2015, pp. 534–541.

[102] K.-Y. Chang, C.-S. Chen, and Y.-P. Hung, "A ranking approach for human ages estimation based on face images," in *Int. Conf. Pattern Recog.*, 2010, pp. 3396–3399.

[103] K. Luu, K. Seshadri, M. Savvides, T. D. Bui, and C. Y. Suen, "Contourlet appearance model for facial age estimation," in *International Joint Conference on Biometrics*. IEEE, 2011.

[104] R. Rothe, R. Timofte, and L. Van Gool, "Some like it hot-visual guidance for preference prediction," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 5553–5561.

[105] W. Shen, K. Zhao, Y. Guo, and A. L. Yuille, "Label distribution learning forests," in *Conf. Neural Inf. Process. Syst.*, 2017, pp. 834–843.

[106] J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 5197–5206.

[107] S. Schulter, C. Leistner, and H. Bischof, "Fast and accurate image upscaling with super-resolution forests," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015, pp. 3791–3799.

[108] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 1646–1654.

[109] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE Conf. Comput. Vis. Pattern Recog.*, vol. 1. IEEE, 2001, pp. I–I.

[110] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 6, pp. 681–685, 2001.

[111] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, 2010.

[112] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Int. Conf. Comput. Vis.*, vol. 2. IEEE, 2001, pp. 416–423.

[113] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. Alberi-Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *Brit. Mach. Vis. Conf.* BMVA press, 2012.

[114] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International Conference on Curves and Surface*. Springer, 2010, pp. 711–730.

[115] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.
[116] H. R. Sheikh, A. C. Bovik, and G. De Veciana, "An information fidelity criterion for image quality assessment using natural scene statistics," *IEEE Trans. Image Process.*, vol. 14, no. 12, pp. 2117–2128, 2005.
[117] E. K. Tang, P. N. Suganthan, and X. Yao, "An analysis of diversity measures," *Machine learning*, vol. 65, no. 1, pp. 247–271, 2006.
[118] J. J. Rodriguez, L. I. Kuncheva, and C. J. Alonso, "Rotation forest: A new classifier ensemble method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 10, pp. 1619–1630, 2006.
[119] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016, pp. 770–778.
[120] S. Zagoruyko and N. Komodakis, "Wide Residual Networks," in *Brit. Mach. Vis. Conf.*, 2016.
[121] Z. Shi, Y. Ye, and Y. Wu, "Rank-based pooling for deep convolutional neural networks," *Neural Networks*, vol. 83, pp. 21–31, 2016.
[122] Y. Liu, M.-M. Cheng, X. Hu, J.-W. Bian, L. Zhang, X. Bai, and J. Tang, "Richer convolutional features for edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1939 – 1946, 2019.
[123] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, "Res2net: A new multi-scale backbone architecture," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.
[124] T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, and M. Li, "Bag of tricks for convolutional neural network," *arXiv preprint arXiv:1812.01187*, 2018.
[125] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2015.