Quality Diversity for Visual Pre-Training

Ruchika Chavhan[1], Henry Gouk[1], Da Li[2], Timothy Hospedales[1,2]
[1]The University of Edinburgh [2]Samsung AI Center, Cambridge
R.Chavhan@sms.ed.ac.uk, {henry.gouk, t.hospedales}@ed.ac.uk,
dali.academic@gmail.com

Abstract

Models pre-trained on large datasets such as ImageNet provide the de-facto standard for transfer learning, with both supervised and self-supervised approaches proving effective. However, emerging evidence suggests that any single pre-trained feature will not perform well on diverse downstream tasks. Each pre-training strategy encodes a certain inductive bias, which may suit some downstream tasks but not others. Notably, the augmentations used in both supervised and self-supervised training lead to features with high invariance to spatial and appearance transformations. This renders them sub-optimal for tasks that demand sensitivity to these factors. In this paper we develop a feature that better supports diverse downstream tasks by providing a diverse set of sensitivities and invariances. In particular, we are inspired by Quality-Diversity in evolution, to define a pre-training objective that requires high quality yet diverse features — where diversity is defined in terms of transformation (in)variances. Our framework plugs in to both supervised and self-supervised pre-training, and produces a small ensemble of features. We further show how downstream tasks can easily and efficiently select their preferred (in)variances. Both empirical and theoretical analysis show the efficacy of our representation and transfer learning approach for diverse downstream tasks. Code available at https://github.com/ruchikachavhan/quality-diversity-pretraining.git

1. Introduction

Pre-training neural networks on large-scale datasets such as ImageNet followed by representation transfer is a dominant paradigm in applied deep learning [58]. Supervised pre-training [34] is long established, with self-supervised [14, 30, 15, 11, 69] pre-training rapidly gaining popularity. Such pre-training algorithms often aspire to providing a universal representation which is effective for diverse downstream tasks across computer vision, with evaluation studies showing promising results [34, 24]. Nevertheless, emerging evidence suggests that any single pre-trained feature will not be optimal for diverse downstream tasks [25, 68, 13]. This is because, any given pre-training strategy encodes a certain inductive bias in the learned representation, which will better suite some downstream tasks more than others.

This issue is evident with regards to transformation invariance. By way of example, downstream object recognition benefits from strong affine invariance in a feature since pose is a distractor for object recognition. In con-

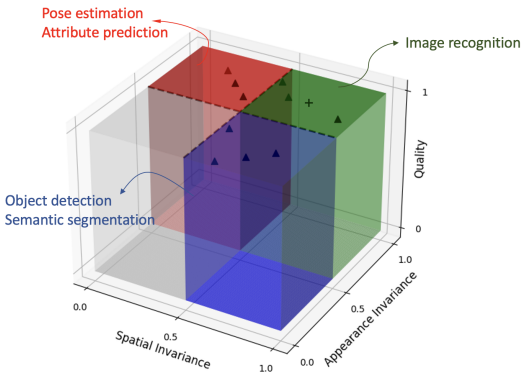

Figure 1. Illustration of the quality and behavior space of QD4V feature representations. Each point in this space is a potential feature extractor. The x and y axes correspond sensitivity-invariance axes for spatial and appearance transformations, and define the behaviour space for features. The z axis shows the quality of the features (eg: ImageNet accuracy). The shaded zones schematically show the range of feature encodings that might be preferred by different types of downstream tasks. Conventional pre-training learns a single solution (denoted with +) that is high-quality (z axis near 1) but by default occupies the green zone of high invariance to spatial and appearance transforms. In contrast, QD4V aims to learn a set of solutions (marked by Δ) that better span the (in)variance axes, while being also being high quality. These lead to better performance for diverse downstream tasks. Note that in practice the invariance space has many more dimensions than the two illustrated here.

trast, downstream pose estimation depends heavily on affine sensitivity in a feature. However, supervised pre-training tends to promote affine-invariance due to the nature of the task and the popular augmentations used. Meanwhile self-supervised pre-training produces feature with even stronger affine-invariance due to optimising for augmentation invariance. So neither may be optimal for downstream pose estimation. More generally, there are a diverse set of potential transformations, each of which may be nuisance factors for some tasks and crucial for others – and any given representation exhibits different different degrees of (in)sensitivity to each each possible transformation. Thus, despite the power of mainstream supervised and self-supervised pre-training, since any *single* feature induces a particular set of (in)variance strengths, it will be sub-optimal for those downstream tasks with different (in)variance preferences.

Within the self-supervised learning community, a few studies have attempted to address these issues by learning features with multiple (in)variances. This often achieved by an ensemble of different models [25], pre-training multi-headed networks with contrastive learning for different augmentations in each head [68], or with different heads dedicated to (invariance promoting) contrastive learning and (sensitivity promoting) transformation prediction [38]. However, these initial attempts suffer important limitations. Firstly, given that there are many transformations for which invariance or sensitivity may be preferred, it's unclear how to build a compact enough ensemble. For example, if N models are used to encode N distinct strengths of sensivity to a particular transformation, and there are K possible transformations, this naively leads to an ensemble of size N^K to cover the possible invariance preferences of downstream tasks. Secondly, it's unclear how to optimally select or fuse these different features for downstream tasks with apriori unknown (in)variance preferences.

In this paper, we address the first of these challenges through the lens of Quality Diversity (QD) [12, 51, 27] a strategy in evolutionary computation that has seen great success in robotics [20]. QD strategies aim to more robustly solve problems by searching for a set of solutions rather than a single solution, and – crucially – requiring that the set of solutions are diverse in some meaningful behavioural metric. We introduce the notion of Quality Diversity Optimisation for Vision (QD4V), where, in a computer vision context, we interpret the idea of behaviour space for diversity measurement as degree of (in)variances to different data augmentations. When applied to supervised ImageNet pre-training, this corresponds to optimising for a small set of ImageNet models that are highly performant, yet use as different cues as possible for recognition. This is qualitatively illustrated in Fig. 1. Each downstream task is then presented with a meaningfully diverse set of high-quality features from which a feature meeting the (in)variance preferences of the task is more likely to be found.

Our second contribution addresses how to fuse this small ensemble of features for the downstream task. The limited existing work on learning multiple (in)variances relies on ad-hoc heuristics, such as simply concatenating features prior to linear readout [25, 68]. We argue that a better solution is train a per feature linear readout on the downstream train set, and then learn a fusion weight on the downstream val set, i.e., stacking [66]. We show both empirically and theoretically that this approach to transfer learning outperforms the standard approach.

In summary, we present a framework for learning a compact and high-quality yet meaningfully diverse set of features. We present a simple approach for transferring them to downstream tasks with strong theoretical backing. Finally, we show empirically that our framework improves on average compared to standard pre-trained features when evaluated on a diverse range of downstream tasks.

2. Related Work

Invariance learning in vision: Invariances for in-domain image classification have been learned using data augmentation [9] via MAP [6] and marginal likelihood [32] learning. However, these methods do not focus on transferability to downstream target tasks. On the other hand, the success of representation transfer in contrastive self-supervised learning has been attributed to augmentation-based training engendering invariances [25, 63, 52] which act as strong inductive biases for downstream tasks. The field aspires to provide a single general-purpose feature suited for all downstream tasks [8], however, this goal is not straightforward to achieve. Recent studies have shown that features with invariances to different augmentations are suited for different downstream tasks, with no single feature being optimal for all tasks [25, 68] and performance suffers if inappropriate invariances are provided. In contrastive learning, this leads to the laborious need to produce and combine an ensemble of features [68, 25], to disentangle invariance and transformation prediction [38], costly task-specific self-supervised pre-training [54, 61], or amortising invariances [13]. We introduces a framework for learning diverse invariances during either supervised or self-supervised pre-training, while maintaining the quality of those features in terms of encoding image semantics. We also provide an efficient and theoretically supported approach to fuse the resulting features that improves on standard heuristics [25, 68].

Quality Diversity: Quality Diversity (QD) optimisation [12, 51, 27] originated in evolutionary algorithms as a strategy for finding a meaningfully diverse set of high-performing solutions to a problem, rather than just one global optimum. Access to a suite of meaningfully diverse high-quality solutions often has substantial robustness benefits that are widely exploited in robotics community

[20]. In order to drive diversity, a QD algorithm requires an application-specific measure of behaviour as a feature between which two models' difference can be compared. We develop a novel behavioural measure for vision in terms of invariances. Mainstream QD optimisation algorithms such as MAP-Elites are not amenable to the differentiable gradient-based solutions required for application in vision. We develop a simple differentiable instantiation of QD that is amenable to easy integration as a loss within a typical deep learning for vision pipeline.

Ensemble methods: QD is related to ensemble methods, which have a long history in vision and machine learning [22]. However, ensembles have mostly focused on single-task learning rather than pre-training for transfer learning. A key challenge in ensemble methods is obtaining meaningful inter-model diversity, with the standard approaches of random seeds, boosting, and bagging, providing comparatively weak diversity. Our novel strategy is to take QD's insight that diversity should be measured in a meaningful space, and identifying (in)variance axes as such a meaningful space for diversity measurement in visual pre-training. A second key issue for ensembles is fusing the ensemble prediction, which has been variously performed by unweighted prediction averaging [45] or parameter averaging [46], and feature concatenation in transfer learning [25, 68]. We show both empirically and theoretically the efficacy of learning a weighted combination of predictions.

3. Methodology

3.1. Background

Mainstream QD optimization [27, 51] searches for a diverse set of solutions $\{\theta_i\}$ to an optimization problem $\mathcal{L}_q$ by solving an objective like

$$\min_{\{\theta_i\}} \sum_i \mathcal{L}_q(\theta_i) \; s.t. \; D(\{\pi(\theta_i)\}), \tag{1}$$

where π is a vector of application relevant observable properties of a given solution θ_i that is called *phenotype* in the QD community. D is a diversity constraint that enforces diversity among the solutions. It is often enforced by tesselation, or gridding, the space of π. Thus solution sets $\{\theta_i\}$ have the maximum posible quality ($\mathcal{L}_q$) while also covering the space of measurable differences in behaviour [27, 51]. To make such QD problems more efficient to solve in a vision context, we will relax them to

$$\min_{\{\theta_i\}} \sum_i \mathcal{L}_q(\theta_i) - \sum_{i \neq j} \mathcal{L}_{div}(\pi(\theta_i), \pi(\theta_j)\}), \tag{2}$$

where $\mathcal{L}_{div}$ is a loss that measures pairwise distance.

3.2. Visual Pre-training

We denote a large dataset for pre-training by $\mathcal{D}_t \subset \mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}$ is some space of input images and the space $\mathcal{Y}$ may be contain labels obtained from manual annotation or pre-text tasks from self-supervision. We introduce a population of N feature extractors, represented as an ensemble, $\mathcal{F} = \{f_{\theta_i}\}_{i=1}^N$, which aim to learn a diverse set of invariances while maintaining high quality. Each member of the ensemble is followed by a projection layer g_{ϕ_i} which can be a classifier layer in case of supervised learning or a projection head in case of self-supervised learning [14, 30, 16]. We denote a set of K augmentations considered for pre-training as $\mathcal{T}$. Let us also denote a sample $\mathbf{x}$ transformed by augmentation $\mathcal{T}_j \in \mathcal{T}$ as $\tilde{\mathbf{x}}^j$.

3.2.1 QD for Visual Pre-training: Diversity

To quantify the diversity of solutions during pre-training, we must first define a phenotype — an observable measurement of a feature extractor's behaviour. We define the behavior of a solution by the degree of invariance it shows to a set of augmentations $\mathcal{T}$. We borrow the definition of invariance from [25], where invariance of a model to a particular data augmentation is defined as the cosine similarity $S(f_\theta(\mathbf{x}), f_\theta(\tilde{\mathbf{x}}^j))$ between augmented and unaugmented features. Thus, we define a phenotype of a solution f_{θ_i} as a vector $\pi(\mathbf{x}; \theta_i) \in \mathbb{R}^K$ such that j^{th} element of the phenotype vector indicates the invariance that the model exhibits to augmentation $\mathcal{T}_j \in \mathcal{T}$, as shown in Eq 3.

$$\pi(\mathbf{x}; \theta_i) = \left(S(f_{\theta_i}(\mathbf{x}), f_{\theta_i}(\tilde{\mathbf{x}}^j) \right)_{j=1}^{|\mathcal{T}|} \tag{3}$$

Diversity Loss: Based on the above phenotype definition, we can encourage each model to exhibit a diverse combination of invariances to augmentations in $\mathcal{T}$. Specifically, we minimise the negative exponential distance between phenotype vector of each member as in Eq 4

$$\mathcal{L}_{\text{diversity}} = \sum_{i \neq j}^{N} \exp(-|\pi(\mathbf{x}; \theta_i) - \pi(\mathbf{x}; \theta_j)|) \tag{4}$$

3.2.2 QD for visual pre-training: Quality

Quality Loss: Depending on the pre-training paradigm employed, different loss functions can be used to promote high quality. In the case of supervised learning, the loss function denoted by l_q can be cross-entropy, while self-supervised methods such as SimCLR/MoCo use contrastive loss [48]. The loss function for the entire ensemble for a sample $(\mathbf{x}, y) \in \mathcal{D}$ is given in Eq. 5

$$\mathcal{L}_{\text{quality}} = \frac{1}{N} \sum_{i=1}^{N} l_q(g_{\phi_i}(f_{\theta_i}(\mathbf{x})), y) \tag{5}$$

This objective reflects standard supervised or self-supervised learning of the upstream model. Empirically,

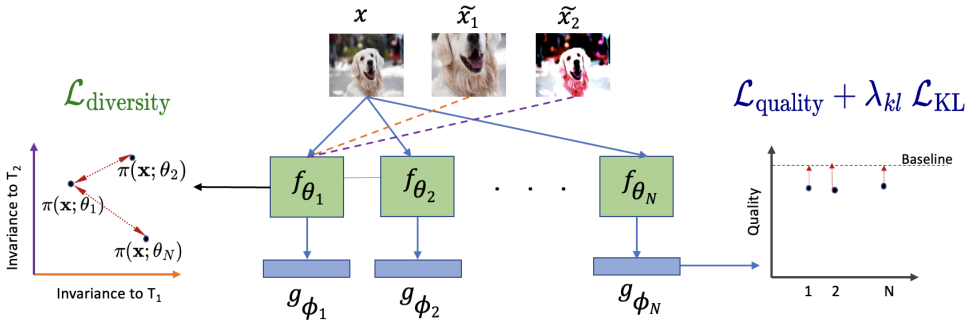

Figure 2. Schematic illustration of Quality Diversity Pre-training, shown with only 2 augmentations for simplicity. $\mathcal{L}_{\text{diversity}}$ maximises the difference between invariances that models exhibit to $\mathcal{T}_1$ and $\mathcal{T}_2$, while $L_{\text{quality}} + \lambda_{kl}\mathcal{L}_{KL}$ maximises the quality of the ensemble by pushing it to the quality of a conventionally pre-trained model (denoted by Baseline).

we found that purely using this quality objective it is hard for a diverse ensemble to match the quality of a baseline model trained solely for quality. To help the diverse ensemble match the quality of the baseline we introduce an additional loss term, which is the Kullback-Leibler (KL) divergence between the posteriors of the ensemble members and baseline model with parameters ϕ_b and θ_b. We denote the predictions of a model by $\hat{y}_i = g_{\phi_i}(f_{\theta_i}(x))$. The loss function is shown in Eq. 6. This term ensures that the members of the ensemble mimic the predictions of the baseline model, thus ensuring high quality with diverse invariances.

$$\mathcal{L}_{\text{quality-KL}} = \sum_{i=1}^{N} \frac{1}{m} \sum_{j=1}^{m} \text{KL}\left(\sigma(\hat{y}_i;\tau) \| \sigma(\hat{y}_b;\tau)\right) \quad (6)$$

Here, σ denotes the softmax function and τ denotes the temperature used to scale the logits within softmax.

Summary Finally, to achieve our goal of diverse invariances and high quality in QD4V pre-training, we formulate a total loss function asin Eq. 7.

$$\mathcal{L}_{qd} = \mathcal{L}_{\text{quality}} + \lambda_{kl}\mathcal{L}_{\text{quality-KL}} + \lambda_d \mathcal{L}_{\text{diversity}} \quad (7)$$

The diverse set of optimal solutions learned from the proposed QD pre-training stage denoted as $\mathcal{F}^*$ is then transfered to a downstream task.

3.3. Downstream task Learning

We next address how to construct models for downstream tasks given the diverse ensemble of features $\mathcal{F}^*$ produced by pre-training in the previous section. Existing approaches have concatenated features prior to linear readout [68], or conducted unweighted averaging of ensemble features [45]. We propose to fuse the ensemble via a variant of stacking [66].

Decoder Design For each feature extractor, $f_{\theta_i} \in \mathcal{F}$, we follow the standard linear readout approach, and build a linear model, $h^{(i)}$, parameterised by $U^{(i)} \in \mathbb{R}^{k\times d}$ and $\mathbf{b}^{(i)} \in \mathbb{R}^k$,

$$h^{(i)}(\mathbf{x}) = \rho(U^{(i)} f_{\theta_i}(\mathbf{x}) + \mathbf{b}^{(i)}),$$

where ρ is a transfer function for linear models (e.g., the softmax function or identity). To produce a final prediction we fuse the ensemble of classifiers $h^{(i)}(x)$ by learning a weight vector, $\mathbf{w} \in \mathbb{R}^N$ that is used to make a linear combination of their predictions,

$$h(\mathbf{x}) = \sum_{i=1}^{N} w_i h^{(i)}(\mathbf{x}).$$

Decoder Training Strategy For each downstream task have a collection of data, $\mathcal{D}_{ds}$, which we split into two non-overlapping subsets, $\mathcal{D}_{tr}$ and $\mathcal{D}_{val}$. The hyperparameters of the linear models are tuned using this train–validation split, and the best model parameters found during this tuning phase are denoted by $\tilde{U}^{(i)}$ and $\tilde{\mathbf{b}}^{(i)}$. Each model is retrained using the full $\mathcal{D}_{ds}$ dataset using the best hyperparameters to obtain the final values for each $U^{(i)}$ and $\mathbf{b}^{(i)}$. The procedure used to fit the weights for linearly combining the models is carried out using the $\tilde{U}^{(i)}$ and $\tilde{\mathbf{b}}^{(i)}$ parameters, and $\mathbf{w}$ is fit on the $\mathcal{D}_{val}$ set,

$$\mathbf{w} = \arg\min_{\mathbf{v}\in\mathbb{R}^N} \mathbb{E}_{\mathcal{D}_{val}} \left[\left\| \sum_{i=1}^{N} v_i \rho(\tilde{U}^{(i)} f_{\theta_i}(\mathbf{x}) + \tilde{\mathbf{b}}^{(i)}) - y \right\|_2^2 \right] \quad (8)$$

In the following section, we provide theoretical justification for our proposed downstream architecture and training strategy by way of bounding the generalisation error for downstream tasks learned in this way.

3.4. Theoretical Analysis

For ease of exposition, consider the case where ρ is the identity and the underlying problem is binary classification.

In this case, each $U^{(i)}$ can be written as a vector, $\mathbf{u}^{(i)}$. We can define the effective hypothesis class of our algorithm for learning on downstream tasks as

$$\mathcal{H} = \Bigg\{ \mathbf{x} \to \sum_{i=1}^{N} w_i \langle \mathbf{u}^{(i)}, f_{\theta_i}(\mathbf{x}) \rangle : \\ \|\mathbf{w} - \mathbb{E}[\mathbf{w}]\|_2 \leq A, \|\mathbf{u}^{(i)} - \mathbb{E}[\mathbf{u}^{(i)}]\|_F \leq B \Bigg\},$$

where we have absorbed the bias values into $\mathbf{u}$ by assuming a 1 has been appended to each $\mathbf{x}$. Our goal is to bound the generalisation error of models from this class trained using our stacking procedure in Sec 3.3. The first step is to bound the empirical Rademacher complexity in terms of A and B (from the definition of $\mathcal{H}$), the norm of the features, and the total amount of data available for the downstream task, $m = |\mathcal{D}_{ds}|$. We then show how the stacking procedure influences the values of A and B. We use η to indicate the fraction of data used for training, hence $\eta m = |\mathcal{D}_{tr}|$ and $(1-\eta)m = |\mathcal{D}_{val}|$. Our main result in given below.

Theorem 1. *For a model trained using our stacking procedure and the conditions outlined in the statements of Lemmas 1 and 2, we have with probability at least* $1 - 2\delta$*,*

$$\begin{aligned} \mathbb{E}[\mathbf{1}[\text{sgn}(h(\mathbf{x})) \neq y]] \leq \hat{\mathbb{E}}[l(h(\mathbf{x}), y)] \\ + \frac{12X^3\|\tilde{V}\|_2\sqrt{\ln(4/\delta)\ln(4N/\delta)}}{\gamma_1\gamma_2\sqrt{1-\eta}m^{3/2}} \\ + \frac{2X^2\sqrt{2\ln(4N/\delta)}}{\gamma_1 m} \\ + \frac{2X^2\|\tilde{V}\|_2\sqrt{2\ln(4/\delta)}}{\gamma_2\sqrt{1-\eta}m} \\ + 3\sqrt{\frac{\ln(1/\delta)}{2m}}, \end{aligned}$$

where $\hat{\mathbb{E}}[l(h(\mathbf{x}), y)]$ *is the mean ramp loss of the model* h*, computed on the training data.*

Notably, this bound achieves the "fast" rate of convergence, $\mathcal{O}(m^{-1})$, in the complexity terms, with the first term even exhibiting a super-fast rate of $\mathcal{O}(m^{-3/2})$. This is in contrast to the standard $\mathcal{O}(m^{-1/2})$ rate in existing bounds. The proof is in the supplemental material.

Discussion This theorem depends on both the specific architecture and optimisation strategy outlined in Sec. 3.3, and thus provides justification for these design choices. It says that the generalisation gap between train error and expected (test) error rapidly goes to zero with the size of the downstream train set m. Furthermore, because we have access to an ensemble of features, the train error $\hat{er}(h)$ can likely be reduced below that of a single feature.

4. Implementation details

Pre-training: We perform supervised training for ResNet50 and ConvNeXt [42] on ImageNet1K [21] and contrastive training for ResNet50 on both ImageNet1K and ImageNet100 [68]. To learn our feature ensemble, in a parameter efficient way for both architectures, we share the first three layers (`layer1`, `layer2`, `layer3` according to [65]), after which different members of the ensemble branch out into their own sequence of layers having a separate `layer4` and projection head. We initialise the entire ensemble with ImageNet pre-trained models in case of supervised pre-training and MoCo pre-trained models in case of contrastive learning.

Augmentations: For supervised pre-training, most methods that rely on strong augmentation policies. Our approach applies the quality loss (Eq. 5) to unaugmented samples only as using augmented samples for quality maximisation would lead to typical strong invariance to all augmentations and prevent diversity. For contrastive learning, we apply very weak augmentations for instance discrimination.

Learning rates and Optimisers: For supervised pre-training and contrastive learning experiments, we train the model for 20 epochs with SGD and AdamW optimizers respectively, along with learning rate warm-up for 5 epochs, followed by a cosine decay schedule. For supervised pre-training, we apply label smoothening and perform exponential moving average (EMA) on the model weights.

Downstream tasks: Our suite of downstream tasks consists of object recognition, regression, and dense estimation. We provide more details about downstream tasks in the supplementary material.

- **Classification:** For classification, we evaluate on standard benchmarks CIFAR10/100 [36], Caltech101 [26], Oxford Flowers [47], Stanford Cars [35], Describable Textures Dataset (DTD) [17], Aircraft [43].
- **Regression:** We include a set of spatially sensitive tasks including facial landmark detection on 300W [1], CelebA [41], human pose estimation in LSP [33] and MPII [2], Animal Pose prediction [10], ALOI object orientation prediction [29], and Causal3DIdent 6D pose and appearance attribute prediction [60].
- **Dense estimation:** We also evaluate the QD pre-trained models on dense estimation tasks like Pascal VOC object detection using Faster R-CNN [55] with a Feature Pyramid Network backbone [39] and linear semantic segmentation similar to [4] on CityScapes and ADE20k [7] datasets.

Downstream evaluation: For classification and regression tasks, we fit multinomial logistic regression and multi-output linear regression on the extracted features from each

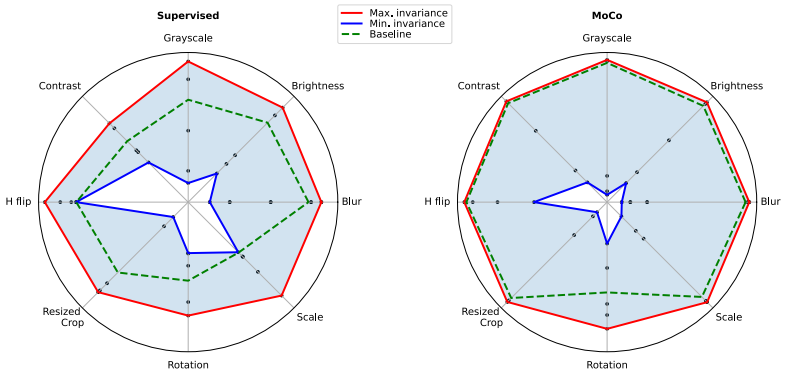

Figure 3. Radar plot comparing invariance w.r.t. for 4 Spatial and 4 Appearance based transformations of baseline models (green) and QD ensemble (dots). The red and blue lines correspond to maximum and minimum invariances within QD ensemble. The blue shaded region corresponds to the range of invariances spanned by the QD ensemble, which encompasses the (in)variance of the baselines.

		Methods	CIFAR10	CIFAR100	Flowers	Caltech 101	DTD	Cars	Aircraft	300w	LS Pose	CelebA	Animal Pose	MPII	ALOI	Causal3D	Rank
IN1K (RN50)	Sup.	Baseline	90.4	68.2	85.2	84.7	71.9	44.4	36.0	70.2	55.4	49.0	11.2	18.0	24.1	64.1	2.7
		Multihead	90.3	67.3	84.8	84.1	69.9	43.2	35.0	68.3	53.2	50.1	10.9	17.0	22.8	62.3	3.7
		Div. Ens. [23]	**91.6**	**72.9**	84.6	89.3	69.8	42.7	35.3	66.6	52.6	49.2	11.5	17.9	22.6	60.8	3.4
		MeTTA [3]	91.0	71.6	85.3	85.1	71.8	41.1	34.0	65.3	52.8	47.9	10.5	16.3	23.1	61.2	3.8
		QD4V	90.3	70.4	**86.9**	**89.9**	**72.2**	**45.4**	**36.9**	**76.6**	**62.4**	**61.5**	**12.5**	**18.5**	**28.4**	**72.8**	**1.4**
	MoCo	MoCo	90.3	70.6	**89.8**	87.9	73.9	39.3	41.8	87.2	69.0	92.5	13.9	19.9	46.0	78.1	1.9
		Fusion^{x} [25]	**95.0**	–	81.0	86.0	–	–	–	65.0	68.0	33.0	–	–	–	–	–
		QD4V	88.9	**71.2**	85.2	**88.2**	**75.8**	**41.3**	**43.9**	**88.9**	**71.8**	**95.2**	**14.2**	**22.4**	**47.4**	**80.6**	**1.2**
IN100	Sup.	Baseline (CN-T)	89.4	70.6	94.5	89.1	68.0	**52.5**	48.4	63.1	48.1	52.5	11.5	17.7	8.02	69.4	1.8
		QD4V (CN-T)	**91.5**	**72.9**	**96.5**	**89.9**	**69.8**	51.5	**50.0**	**82.2**	**69.9**	**60.7**	**12.5**	**18.5**	**9.3**	**71.9**	**1.1**
	MoCo	MoCo	84.6	61.6	82.4	77.3	64.5	33.9	37.2	85.5	58.7	61.0	13.2	18.6	30.9	61.4	3.5
		AugSelf* [38]	**85.3**	63.9	**85.7**	78.9	66.2	37.4	**39.5**	77.3	63.9	77.0	12.9	19.5	**35.2**	61.6	2.4
		AI^{+} [13]	81.3	64.6	81.3	78.4	68.8	38.6	37.3	**90.0**	65.2	82.0	12.5	**21.6**	32.7	62.6	2.5
		QD4V	84.5	**65.4**	85.5	**81.2**	**71.9**	**39.8**	37.9	88.9	**69.7**	**85.0**	**14.3**	20.6	33.8	**65.3**	**1.5**

Table 1. Downstream performance of ImageNet (IN1K) pretrained (1) Supervised ResNet50, (2) MoCo ResNet50 and ImageNet-100 (IN100) pretrained (3) Supervised ConvNeXt-Tiny (CN-T), (4) MoCo ResNet50. The first seven columns are classification tasks (accuracy, %); the last seven are regression (R2, %). $^{+}$ numbers from [13] where reported, * A mix of our runs and numbers from [13] where reported, ^{x} numbers from [25], rest are our runs.

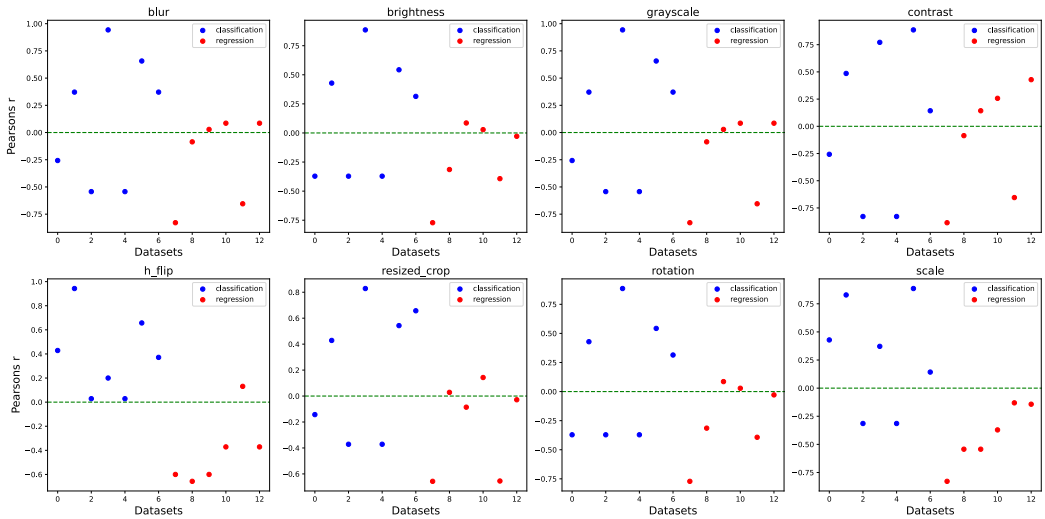

Figure 4. Correlation between our classifier fusion weights (Eq 8) and task for 4 Spatial and 4 Appearance based augmentations. The y-axis corresponds to Pearson's r correlation between $\mathbf{w}$ assigned to each member by a particular task and invariances exhibited by that member to specific augmentations. Tasks are grouped by classification and regression, which tend to show different invariance preferences.

Methods	CUB		Flowers		FC 100		Plant Disease		300w		LS Pose		CelebA		Causal3D		Rank
	(5, 1)	(5, 5)	(5, 1)	(5, 5)	(5, 1)	(5, 5)	(5, 1)	(5, 5)	$s = 0.05$	$s = 0.2$	$s = 0.05$	$s = 0.2$	$s = 0.05$	$s = 0.2$	$s = 0.05$	$s = 0.2$	
Baseline	70.0	90.4	77.3	93.7	53.8	**78.7**	68.9	88.8	20.5	24.8	46.9	48.5	39.2	41.4	58.9	62.4	2.3
Multihead	69.6	88.2	76.5	92.6	52.9	78.5	68.2	87.3	19.4	24.7	39.3	44.9	39.4	40.8	57.2	61.3	3.5
Div. Ensemble	68.3	88.7	**78.4**	**94.6**	54.9	78.1	68.7	**91.9**	18.1	18.8	35.7	36.8	39.3	39.7	54.7	58.9	3.4
MeTTA	68.4	88.9	77.6	92.3	54.4	72.8	68.7	87.5	18.3	20.3	36.8	37.5	37.2	37.9	49.3	58.6	4.1
Ours	**71.4**	**90.8**	73.4	93.2	**57.5**	76.2	**69.4**	91.0	**37.5**	**44.9**	**59.7**	**60.3**	**50.5**	**53.1**	**60.3**	**63.4**	**1.6**
MoCo	**51.7**	69.8	**71.8**	89.4	**54.9**	71.0	**67.5**	85.3	39.9	48.3	59.3	63.1	47.5	59.0	66.7	77.4	1.8
Ours	49.9	**74.1**	70.4	**93.4**	50.9	**72.9**	62.3	**89.6**	**57.3**	**60.4**	**62.6**	**65.6**	**64.2**	**64.7**	**70.3**	**79.4**	**1.3**
MoCo	41.0	56.9	69.6	76.4	31.7	43.9	65.7	85.0	39.0	50.1	54.2	60.3	40.2	52.3	52.5	57.0	3.7
AugSelf [38]	44.2	57.4	76.0	85.6	35.0	48.8	71.8	87.8	42.0	51.8	53.8	60.1	53.2	66.3	54.4	61.8	2.7
LooC [38]	-	-	70.9	80.8	-	-	-	-	-	-	-	-	-	-	–	–	–
AI [13]	**45.0**	**58.0**	**76.7**	88.7	37.4	48.4	**72.6**	**89.1**	49.2	**57.9**	55.3	62.0	**56.0**	**76.0**	60.2	62.8	**1.6**
Ours	41.3	56.6	67.8	**88.9**	**39.5**	**53.7**	65.9	88.1	**51.1**	57.5	**58.3**	**64.8**	50.2	57.7	**61.8**	**63.9**	1.9

Table 2. Few-shot classification and regression accuracy (%, R2) of our QD ResNet50 ensemble for supervised (top), MoCo pre-trained on ImageNet (middle) and MoCo pre-trained on ImageNet-100 (bottom). Values are reported with 95% confidence intervals averaged over 2000 episodes on FC100, CUB200, and Plant Disease. (N, K) denotes N-way K-shot tasks. For regression tasks (300w, LS Pose, CelebA, Causal3D), we report downstream performance for different splits with train proportion given by s.

of the ensemble and then learn a weighted combination of over all these predictions using linear regression. We sweep the regularisation parameters for all members for each downstream dataset based on its validation set. For dense tasks, we train a decoder head for each member of the ensemble using default hyperparameters in [67] for object detection and [18] for semantic segmentation.

Few-shot downstream tasks: We also evaluate the pre-trained networks on various few-shot learning benchmarks: FC100 [49], Caltech-UCSD Birds (CUB200), and Plant Disease [44]. We also show results for few-shot regression problems on 300w, Leeds Sports Pose, CelebA, and Causal3DIdent datasets, where we we repeatedly sampled 5%, and 20% to generate low-shot training sets. For few-shot classification, we perform logistic regression using the frozen ensemble and learn episode-wise fusion weights $\mathbf{w}$ the support set itself.

Competitors: All methods are evaluated by readouts on fixed features. **Supervised:** Besides conventional supervised pre-training (denoted 'baseline'), we also train a conventional ensemble via a multi-head model ('multihead') that has the same architecture and number of parameters as our architecture but uses only $\mathcal{L}_q$ loss. We also compare with another diverse ensemble strategy [23, 50] (denoted 'div ensemble'): to optimises for diversity in terms of KL between posterior probability of ensemble members. The idea being that differences in secondary probabilities are a good measure of diversity. We provide more details in the supplementary material. Finally, we compare with a test-time ensemble created by taking a single pre-trained model and generating different invariances through test-time mean embeddings [3] (denoted 'MeTTA'). **Contrastive:** We compare our approach to two state of the art ensemble based alternatives AugSelf [68], LOOC [38] and amortised invariances (AI) [13]. Note that unlike other competitors, AI uses backprop to update the features.

	Methods	CityScapes		ADE20k		Pascal VOC			Rank
		MIoU	Acc.	MIoU	Acc.	AP	AP50	AP75	
Sup.	Baseline	39.6	82.1	30.4	52.9	48.8	80.3	51.6	2.6
	Multihead	35.4	80.3	27.3	51.8	49.8	80.2	50.9	3.9
	Div. Ensemble	36.2	**82.4**	27.6	52.7	50.6	81.9	58.1	2.3
	QD4V	**41.0**	81.7	**33.8**	**54.3**	**50.8**	**82.5**	**58.7**	**1.3**
	Baseline (FT)	61.7	94.1	40.1	67.2	55.1	82.8	60.5	2.0
	QD4V (FT)	**65.2**	**94.3**	**42.3**	**67.4**	**56.2**	**84.9**	**62.9**	**1.0**
MoCo.	Baseline	46.3	87.4	35.2	50.4	**54.2**	81.8	**59.9**	1.7
	QD4V	**49.2**	**88.9**	**39.0**	**54.6**	53.3	**82.2**	**59.9**	**1.1**
	Baseline	34.5	83.4	26.7	40.6	41.2	71.4	42.9	3.0
	AugSelf	**37.9**	**84.9**	27.3	40.8	43.2	73.4	**48.1**	1.6
	QD4V	36.1	**84.9**	**28.5**	**41.7**	**45.9**	**75.2**	**48.0**	**1.3**

Table 3. Downstream performance of (1) Frozen supervised (2) fine-tuned supervised, (3/4) MoCO trained on ImageNet1K/100 model evaluated on semantic segmentation for CityScapes and ADE20k datasets, and Pascal VOC object detection.

5. Results

Can we successfully learn diverse (in)variance strengths with QD pre-training? To answer this question, we use the definition of invariance in Eq. 3 and evaluate the invariances learned by both default models (i.e., supervised and MoCo) and members of our QD ensemble for four appearance augmentations (blur, brightness, grayscale, contrast) and four spatial augmentations (resized crop, horizontal flip, rotation, scale). We compare the invariance strengths learned by the models via the visualization shown in Figure 3. The results show that the standard models provide a fixed set of invariance strengths (green). The members of the QD ensemble (dots) span a diverse range of invariance strengths from low (blue) to high-invariance (red), which can be selected downstream by picking among ensemble members.

Does QD4V benefit a set of downstream tasks with diverse invariance requirements? Our suite of downstream tasks is represent a diverse range of objectives with differing invariance-sensitivity requirements. The ability to exploit the appropriate (in)variance for each task is crucial to achieving high performance. As discussed earlier, off-the-shelf models tend towards high-invariance to multiple factors (Figure 3), which is often effective for object

	Decoder	Aircraft	Flowers	DTD	Animal Pose	ALOI	Causal3D	Rank
Sup.	Concat	36.8	85.9	70.9	11.2	26.6	72.9	2.0
	Average	33.8	83.4	69.1	9.1	23.7	**74.3**	2.7
	Ours	**36.9**	**86.9**	**72.2**	**12.5**	**28.4**	72.8	**1.3**
MoCo	Concat	42.0	83.5	72.9	13.7	45.7	80.2	2.2
	Average	37.9	83.9	72.0	12.6	42.9	78.1	2.8
	Ours	**43.9**	**85.2**	**75.8**	**14.2**	**47.4**	**80.6**	**1.0**
MoCo.	Concat	35.9	84.9	68.6	13.8	28.2	65.1	2.0
	Average	33.3	77.8	64.6	12.1	25.3	65.0	3.0
	Ours	**37.9**	**85.5**	**71.9**	**14.3**	**33.8**	**65.3**	**1.0**

Table 4. Comparison of our proposed downstream task decoder with conventional approaches. Top: Supervised/ImageNet 1k pre-trained. Middle/Bottom: Moco pre-training on ImageNet 1K/100. Our fusion strategy performs better than learning a classifier over concatenated features and averaging features of the ensemble.

recognition, but may not be effective for fine-grained tasks like CelebA facial attributes, or pose and lighting-prediction tasks like 300W and ALOI/Causal3D.

To evaluate the ability of features to support diverse downstream tasks, we evaluate the competitors on a range of tasks likely to exhibit different invariance preferences in Table 1. Our approach achieves comparable or better performance than baselines on all tasks against both supervised and self-supervised baselines. It has particularly remarkable improvements in regression tasks such as CelebA and Causal3D where we see an improvement of 12.5% and 8.5% over supervised pre-training of ResNet50.

To better understand the success of our method, we analyse the weights assigned to each member of the QD ensemble by different downstream tasks. Specifically, we employ Pearson's r correlation coefficient between $\mathbf{w}$ and the invariances learned by the QD ensemble (in Fig 3) as a metric to measure the strength of the relationship between the two variables. The resulting correlations are shown in Figure 4 and grouped by classification vs regression tasks. The trend shows that the spatially sensitive regression tasks tend to assign lower or negative weights to spatially invariant models, whereas classification tasks prefer models with generally stronger invariances. This analysis illustrates how our framework achieves reliably high performance across the range of tasks evaluated quantitatively in Table 1 – by enabling task-specific invariance preference selection.

Does QD4V benefit few-shot learning tasks? To answer this question we focused on MoCo-v2 CNN models trained on ImageNet1K and ImageNet100. For classification tasks, we followed [38, 68] in sampling C-way K-shot epsiodes from the target problem and training linear readouts and learning $\mathbf{w}$ for each episode. For regression tasks we repeatedly sampled 5%, and 20% to generate low-shot training sets. From the results in Table 2, QD pre-training usually performs better than all competitors, with substantial margins in several cases, especially for pose estimation tasks.

Does QD4V benefit dense estimation tasks? We evaluate QD pre-training on semantic segmentation on Cityscapes [19] and ADE20k [7] datasets. We evaluate QD4V by (1) Freezing the entire backbone and solely train a linear segmentation head and (2) Finetuning the entire backbone. We report the Mean Intersection over Union (IoU) and average accuracy for segmentation. Similarly, we also examine the performance of QD pre-training on Pascal VOC object detection. The results for semantic segmentation and object detection for frozen features and finetuning in Table 3 show that QD pre-training generally provides performance gains.

Ablation study for decoders: Finally, we compare our proposed downstream task decoder (Section 3.3) with conventional approaches. The results in Table 4 show that learning fusion weights using Eq. 8 consistently performs better, especially for regression tasks than conventional concatenation and prediction averaging. Intuitively, this is because our stacked fusion decoder has the chance for explicit invariance weighting. More technically, this is explained by the fast convergence rate in Theorem 3.4 which shows that our generalisation gap (between train and test error) tends to zero much faster than standard linear models as a function of the amount of training data. (See Supplementary for further discussion).

How does QD4V compare to large scale pre-trained models? The seminal CLIP [53] model is highly effective at various downstream tasks. Table 9 (supplementary) compares CLIP/RN50 to ImageNet1K QD4V/RN50 in terms of invariances and performance on downstream pose sensitive tasks. The results show that QD4V substantially outperforms CLIP, despite being trained on orders of magnitude less data - which we attribute to CLIP having learned too strong spatial and appearance invariances. This shows that pre-training scale is not a substitute for providing the correct invariance for the downstream task at hand.

6. Conclusion

We addressed the difficulty of pre-training a model to achieve high performance on diverse downstream tasks by introducing the notion of quality-diversity pre-training. By instantiating diversity optimisation in terms of the (in)variance properties of a feature extractor, we obtain a compact set of high-quality features with diverse (in)variance properties. Together with an efficient and effective fusion strategy, we show strong performance on a diverse range of downstream tasks including classification, pose estimation, semantic segmentation, and object detection. Our approach presents a step towards the vision of a universal feature representation capable of supporting multiple vision tasks. In future work we will explore other definitions of of behavioural phenotypes in QD4V.

Acknowledgements This research was partially supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/S000631/1 and the MOD University Defence Research Collaboration (UDRC) in Signal Processing.

References

[1] 300 faces in-the-wild challenge: database and results. *Image and Vision Computing*, 2016. 300-W, the First Automatic Facial Landmark Detection in-the-Wild Challenge. 5, 13

[2] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *CVPR*, 2014. 5, 13

[3] Arsenii Ashukha, Andrei Atanov, and Dmitry Vetrov. Mean embeddings with test-time data augmentation for ensembling of representations. *ICML Workshop on Uncertainty and Robust- ness in Deep Learning*, 2021. 6, 7

[4] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicregl: Self-supervised learning of local visual features. *NIPS*, 2022. 5, 14

[5] Peter L. Bartlett and Shahar Mendelson. Rademacher and Gaussian Complexities: Risk Bounds and Structural Results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002. 19, 21

[6] Gregory W Benton, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson. Learning invariances in neural networks. In *NIPS*, 2020. 2

[7] Xavier Puig Sanja Fidler Adela Barriuso Bolei Zhou, Hang Zhao and Antonio Torralba. The amsterdam library of object images. *CVPR*, 2017. 5, 8

[8] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *ArXiv*, 2021. 2

[9] Aleksander Botev, Matthias Bauer, and Soham De. Regularising for invariance to data augmentation improves supervised learning. *arXiv*, 2022. 2

[10] Jinkun Cao, Hongyang Tang, Hao-Shu Fang, Xiaoyong Shen, Cewu Lu, and Yu-Wing Tai. Cross-domain adaptation for animal pose estimation. In *ICCV*, 2019. 5, 13

[11] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *NIPS*, 2020. 1

[12] Konstantinos Chatzilygeroudis, Antoine Cully, Vassilis Vassiliades, and Jean-Baptiste Mouret. Quality-diversity optimization: a novel branch of stochastic optimization. In *Black Box Optimization, Machine Learning, and No-Free Lunch Theorems*. 2021. 2

[13] Ruchika Chavhan, Jan Stuehmer, Calum Heggan, Mehrdad Yaghoobi, and Timothy Hospedales. Amortised invariance learning for contrastive self-supervision. In *ICLR*, 2023. 1, 2, 6, 7

[14] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 1, 3

[15] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *ArXiv*, 2020. 1, 12

[16] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. *ICCV*, 2021. 3

[17] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *CVPR*, 2014. 5, 13

[18] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020. 7, 14

[19] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 8

[20] Antoine Cully, Jeff Clune, Danesh Tarapore, and Jean-Baptiste Mouret. Robots that can adapt like animals. *Nature*, 2015. 2, 3

[21] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 5

[22] Thomas Dietterich. Ensemble methods in machine learning. In *Multiple Classifier Systems*, Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2000. 3

[23] N. Dvornik, J. Mairal, and C. Schmid. Diversity with cooperation: Ensemble methods for few-shot classification. In *ICCV*, 2019. 6, 7, 12

[24] Linus Ericsson, Henry Gouk, and Timothy M Hospedales. How well do self-supervised models transfer? In *CVPR*, 2021. 1

[25] Linus Ericsson, Henry Gouk, and Timothy M. Hospedales. Why do self-supervised models transfer? investigating the impact of invariance on downstream tasks. *BMVC*, 2022. 1, 2, 3, 6

[26] Li Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *CVPR*, 2004. 5, 13

[27] Matthew Christopher Fontaine and Stefanos Nikolaidis. Differentiable quality diversity. In *NIPS*, 2021. 2, 3

[28] Geoff French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. In *ICLR*, 2018. 14

[29] Burghouts G.J. Smeulders A.W. Geusebroek, JM. The amsterdam library of object images. *IJCV*, 2005. 5, 13

[30] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 1, 3

[31] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *ICCV*, 2021. 14

[32] Alexander Immer, Tycho FA van der Ouderaa, Vincent Fortuin, Gunnar Rätsch, and Mark van der Wilk. Invariance learning in deep neural networks with differentiable laplace approximations. *NIPS*, 2022. 2

[33] Sam Johnson and Mark Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *BMVC*. Aberystwyth, UK, 2010. 5, 13

[34] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? In *CVPR*, 2019. 1

[35] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCV Workshop on 3D Representation and Recognition (3dRR-13)*, 2013. 5, 13

[36] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Technical report, University of Toronto*, 2009. 5, 13

[37] Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution, 2022. 14

[38] Hankook Lee, Kibok Lee, Kimin Lee, Honglak Lee, and Jinwoo Shin. Improving transferability of representations via augmentation-aware self-supervision. *NIPS*, 2021. 2, 6, 7, 8, 12

[39] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. *CVPR*, 2016. 5, 13

[40] Tongliang Liu, Gábor Lugosi, Gergely Neu, and Dacheng Tao. Algorithmic Stability and Hypothesis Complexity. In *Proceedings of the 34th International Conference on Machine Learning*, pages 2159–2167. PMLR, July 2017. 21

[41] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 5, 13

[42] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *CVPR*, 2022. 5

[43] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. Technical report, 2013. 5, 13

[44] Sharada P Mohanty, David P Hughes, and Marcel Salathé. Using deep learning for image-based plant disease detection. *Frontiers in plant science*, 2016. 7

[45] João Moreira, Carlos Soares, Alípio Jorge, and Jorge Sousa. Ensemble approaches for regression: A survey. *ACM Computing Surveys*, 2012. 3, 4

[46] Ury Naftaly, Nathan Intrator, and David Horn. Optimal ensemble averaging of neural networks. *Network: Computation in Neural Systems*, 1997. 3

[47] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *ICCVGIP*, 2008. 5, 13

[48] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *Arxiv*, 2018. 3

[49] Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. *NIPS*, 2018. 7

[50] Tianyu Pang, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. Improving adversarial robustness via promoting ensemble diversity. In *ICML*, 2019. 7, 12

[51] Justin Pugh, Lisa Soros, and Kenneth Stanley. Quality diversity: A new frontier for evolutionary computation. *Frontiers in Robotics and AI*, 2016. 2, 3

[52] Senthil Purushwalkam and Abhinav Gupta. Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases. *NIPS*, 2020. 2

[53] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. 8, 15

[54] Aniruddh Raghu, Jonathan Lorraine, Simon Kornblith, Matthew McDermott, and David K Duvenaud. *NIPS*, 2021. 2

[55] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 5, 13

[56] Shibani Santurkar, Dimitris Tsipras, and Aleksander Madry. Breeds: Benchmarks for subpopulation shift. In *ICLR 2021*, 2020. 14

[57] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014. 20

[58] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. A survey on deep transfer learning. In *ICANN*, 2018. 1

[59] Shuhan Tan, Xingchao Peng, and Kate Saenko. Class-imbalanced domain adaptation: An empirical odyssey. In Adrien Bartoli and Andrea Fusiello, editors, *ECCV*, 2020. 14

[60] Julius von Kügelgen, Yash Sharma, Luigi Gresele, Wieland Brendel, Bernhard Schölkopf, Michel Besserve, and Francesco Locatello. Self-supervised learning with data augmentations provably isolates content from style. *NIPS*, 2021. 5, 13

[61] Diane Wagner, Fabio Ferreira, Danny Stoll, Robin Tibor Schirrmeister, Samuel Müller, and Frank Hutter. On the importance of hyperparameters and data augmentation for self-supervised learning. *ICML Pre-training Workshop*, 2022. 2

[62] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *NIPS*. Curran Associates, Inc., 2019. 14

[63] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. *ICML*, 2020. 2

[64] Andre Wibisono, Lorenzo Rosasco, and Tomaso Poggio. Sufficient Conditions for Uniform Stability of Regularization Algorithms. Technical report, MIT, Dec. 2009. 21

[65] Ross Wightman. Pytorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019. 5

[66] David H. Wolpert. Stacked generalization. *Neural Networks*, 1992. 2, 4

[67] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019. 7, 14

[68] Tete Xiao, Xiaolong Wang, Alexei A Efros, and Trevor Darrell. What should not be contrastive in contrastive learning. In *ICLR*, 2021. 1, 2, 3, 4, 5, 7, 8

[69] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *ICML*, 2021. 1