VIEWMAKER NETWORKS: LEARNING VIEWS FOR UNSUPERVISED REPRESENTATION LEARNING

Alex Tamkin, Mike Wu, Noah Goodman
Department of Computer Science
Stanford University
Stanford, CA 94305, USA
{atamkin, wumike, ngoodman}@stanford.edu

ABSTRACT

Many recent methods for unsupervised representation learning train models to be invariant to different "views," or distorted versions of an input. However, designing these views requires considerable trial and error by human experts, hindering widespread adoption of unsupervised representation learning methods across domains and modalities. To address this, we propose *viewmaker networks*: generative models that learn to produce useful views from a given input. Viewmakers are *stochastic bounded adversaries*: they produce views by generating and then adding an ℓ_p-bounded perturbation to the input, and are trained adversarially with respect to the main encoder network. Remarkably, when pretraining on CIFAR-10, our learned views enable comparable transfer accuracy to the well-tuned SimCLR augmentations—despite not including transformations like cropping or color jitter. Furthermore, our learned views significantly outperform baseline augmentations on speech recordings (+9 points on average) and wearable sensor data (+17 points on average). Viewmaker views can also be combined with handcrafted views: they improve robustness to common image corruptions and can increase transfer performance in cases where handcrafted views are less explored. These results suggest that viewmakers may provide a path towards more general representation learning algorithms—reducing the domain expertise and effort needed to pretrain on a much wider set of domains. Code is available at https://github.com/alextamkin/viewmaker.

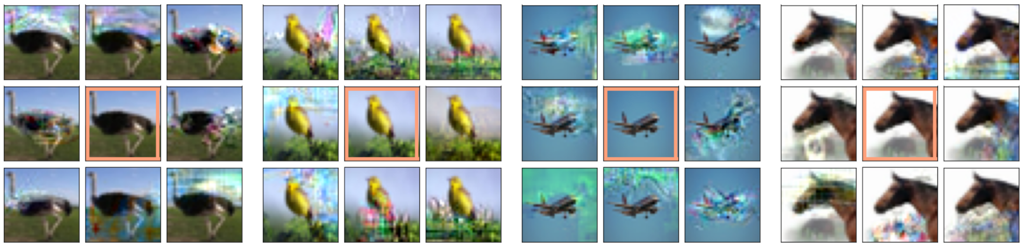


Figure 1: **Viewmaker networks generate complex and diverse input-dependent views for unsupervised learning.** Examples shown are for CIFAR-10. Original image in center with pink border.

1 INTRODUCTION

Unsupervised representation learning has made significant recent strides, including in computer vision, where view-based methods have enabled strong performance on benchmark tasks (Wu et al., 2018; Oord et al., 2018; Bachman et al., 2019; Zhuang et al., 2019; Misra & Maaten, 2020; He et al., 2020; Chen et al., 2020a). *Views* here refer to human-defined data transformations, which target capabilities or invariances thought to be useful for transfer tasks. In particular, in contrastive learning of visual representations, models are trained to maximize the mutual information between different views of an image, including crops, blurs, noise, and changes to color and contrast (Bachman et al., 2019; Chen et al., 2020a). Much work has investigated the space of possible image views (and their compositions) and understanding their effects on transfer learning (Chen et al., 2020a; Wu et al., 2020; Tian et al., 2019; Purushwalkam & Gupta, 2020).

The fact that views must be hand designed is a significant limitation. While views for image classification have been refined over many years, new views must be developed from scratch for new modalities. Making matters worse, even *within* a modality, different domains may have different optimal views (Purushwalkam & Gupta, 2020). Previous studies have investigated the properties of good views through the lens of mutual information (Tian et al., 2020; Wu et al., 2020), but a broadly-applicable approach for learning views remains unstudied.

In this work, we present a general method for learning diverse and useful views for contrastive learning. Rather than searching through possible compositions of existing view functions (Cubuk et al., 2018; Lim et al., 2019), which may not be available for many modalities, our approach produces views with a generative model, called the *viewmaker* network, trained jointly with the encoder network. This flexibility enables learning a broad set of possible view functions, including input-dependent views, without resorting to hand-crafting or expert domain knowledge. The viewmaker network is trained adversarially to create views which increase the contrastive loss of the encoder network. Rather than directly outputting views for an image, the viewmaker instead outputs a stochastic perturbation that is *added* to the input. This perturbation is projected onto an ℓ_p sphere, controlling the effective strength of the view, similar to methods in adversarial robustness. This constrained adversarial training method enables the model to reduce the mutual information between different views while preserving useful input features for the encoder to learn from.

In summary, we contribute:

1. Viewmaker networks: to our knowledge the first modality-agnostic method to *learn* views for unsupervised representation learning
2. On image data, where expert-designed views have been extensively optimized, our viewmaker-models achieve comparable transfer performance to state of the art contrastive methods while being more robust to common corruptions.
3. On speech data, our method significantly outperforms existing human-defined views on a range of speech recognition transfer tasks.
4. On time-series data from wearable sensors, our model significantly outperforms baseline views on the task of human activity recognition (e.g., cycling, running, jumping rope).

2 Related work

Unsupervised representation learning Learning useful representations from unlabeled data is a fundamental problem in machine learning (Pan & Yang, 2009; Bengio et al., 2013). A recently successful framework for unsupervised representation learning for images involves training a model to be invariant to various data transformations (Bachman et al., 2019; Misra & Maaten, 2020), although the idea has much earlier roots (Becker & Hinton, 1992; Hadsell et al., 2006; Dosovitskiy et al., 2014). This idea has been expanded by a number of contrastive learning approaches which push embeddings of different views, or transformed inputs, closer together, while pushing other pairs apart (Tian et al., 2019; He et al., 2020; Chen et al., 2020a;b;c), as well as non-contrastive approaches which do not explicitly push apart unmatched views (Grill et al., 2020; Caron et al., 2020). Related but more limited setups have been explored for speech, where data augmentation strategies are less explored (Oord et al., 2018; Kharitonov et al., 2020).

Understanding and designing views Several works have studied the role of views in contrastive learning, including from a mutual-information perspective (Wu et al., 2020), in relation to specific transfer tasks (Tian et al., 2019), with respect to different kinds of invariances (Purushwalkam & Gupta, 2020), or via careful empirical studies (Chen et al., 2020a). Outside of a contrastive learning framework, Gontijo-Lopes et al. (2020) study how data augmentation aids generalization in vision models. Much work has explored different handcrafted data augmentation methods for supervised learning of images (Hendrycks et al., 2020; Lopes et al., 2019; Perez & Wang, 2017; Yun et al., 2019; Zhang et al., 2017), speech (Park et al., 2019; Kovács et al., 2017; Tóth et al., 2018; Kharitonov et al., 2020), or in feature space (DeVries & Taylor, 2017).

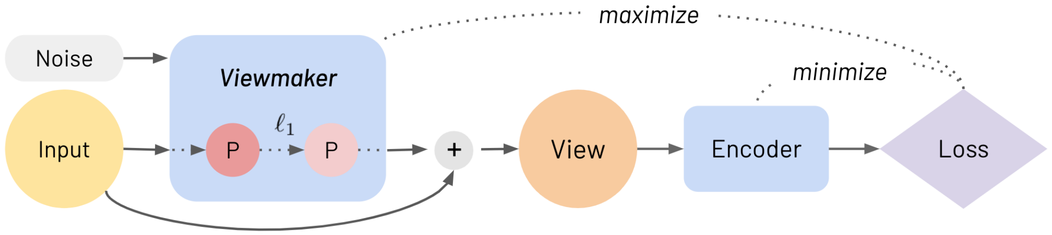


Figure 2: **Diagram of our method.** The viewmaker network is trained to produce stochastic adversarial views restricted to an ℓ_1 sphere around the input.

Adversarial methods Our work is related to and inspired by work on adversarial methods, including the ℓ_p balls studied in adversarial robustness (Szegedy et al., 2013; Madry et al., 2017; Raghunathan et al., 2018) and training networks with adversarial objectives (Goodfellow et al., 2014; Xiao et al., 2018). Our work is also connected to the vicinal risk minimization principle (Chapelle et al., 2001) and can be interpreted as producing amortized virtual adversarial examples (Miyato et al., 2018). Previous adversarial view-based pretraining methods add adversarial noise on top of existing handcrafted views (Kim et al., 2020) or require access to specific transfer tasks during pretraining (Tian et al., 2020). In contrast, our method is more general: it is neither specialized to a particular downstream task, nor requires neither human-defined view families. Outside of multi-view learning paradigms, adversarial methods have also seen use for representation learning in GANs (Donahue et al., 2016; Donahue & Simonyan, 2019) or in choosing harder negative samples (Bose et al., 2018), as well as for data augmentation (Antoniou et al., 2017; Volpi et al., 2018; Bowles et al., 2018). Adversarial networks that perturb inputs have also been investigated to improve GAN training (Sajjadi et al., 2018) and to remove "shortcut" features (e.g., watermarks) for self-supervised pretext tasks (Minderer et al., 2020).

Learning views Outside of adversarial approaches, our work is related to other studies that seek to learn data augmentation strategies by composing existing human-designed augmentations (Ratner et al., 2017; Cubuk et al., 2018; Zhang et al., 2019; Ho et al., 2019; Lim et al., 2019; Cubuk et al., 2020) or by modeling variations specific to the data distribution (Tran et al., 2017; Wong & Kolter, 2020). By contrast, our method requires no human-defined view functions, does not require first pretraining a generative model, and can generate perturbations beyond naturally-occurring variation observed in the training data (e.g. brightness or contrast), potentially conferring robustness benefits, as we explore in Section 4.3.

3 Method

In contrastive learning, the objective is to push embeddings of positive views (derived from the same input) close together, while pushing away embeddings of negative views (derived from different inputs). We focus mainly on the simple, yet performant, SimCLR contrastive learning algorithm (Chen et al., 2020a), but we also consider a memory bank-based algorithm (Wu et al., 2018) in Section 4. As our method is agnostic to the specific pretraining loss used, it is naturally compatible with other view-based algorithms such as MoCo (He et al., 2020), BYOL (Grill et al., 2020), and SwAV (Caron et al., 2020) by similarly substituting the data transformation pipeline with a viewmaker network.

Formally, given a batch of N pairs of positive views (i, j) the SimCLR loss is

$$\mathcal{L} = \frac{1}{2N}\sum_{k=1}^{N}[\ell(2k-1, 2k) + \ell(2k, 2k-1)] \;\text{ where }\; \ell(i,j) = -\log\frac{\exp(s_{i,j}/\tau)}{\sum_{k=1}^{2N}\mathbb{1}_{[k\neq i]}\exp(s_{i,k}/\tau)}$$

and $s_{a,b}$ is the cosine similarity of the embeddings of views a and b.

We generate views by perturbing examples with a *viewmaker* network V, trained jointly with the main *encoder* network M. There are three attributes desirable for useful perturbations, each of which motivates an aspect of our method:

1. **Challenging:** The perturbations should be complex and strong enough that an encoder must develop useful representations to perform the self-supervised task. We accomplish this by generating perturbations with a neural network that is trained adversarially to increase the loss of the encoder network. Specifically, we use a neural network that ingests the input X and outputs a view $X + V(X)$.
2. **Faithful:** The perturbations must not make the encoder task impossible, being so strong that they destroy all features of the input. For example, perturbations should not be able to zero out the input, making learning impossible. We accomplish this by constraining the perturbations to an ℓ_p sphere around the original input. ℓ_p constraints are common in the adversarial robustness literature where perturbations are expected to be indistinguishable. In our experiments, we find the best results are achieved with an ℓ_1 sphere, which grants the viewmaker a *distortion budget* that it can spend on a small perturbation for a large part of the input or a more extreme perturbation for a smaller portion.
3. **Stochastic:** The method should be able to generate a variety of perturbations for a single input, as the encoder objective requires contrasting two different views of an input against each other. To do this, we inject random noise into the viewmaker, such that the model can learn a stochastic function that produces a different perturbed input each forward pass.

Figure 2 summarizes our method. The encoder and viewmaker are optimized in alternating steps to minimize and maximize $\mathcal{L}$, respectively. We use an image-to-image neural network as our viewmaker network, with an architecture adapted from work on style transfer (Johnson et al., 2016). See the Appendix for more details. This network ingests the input image and outputs a perturbation that is constrained to an ℓ_1 sphere. The sphere's radius is determined by the volume of the input tensor times a hyperparameter ϵ, the *distortion budget*, which determines the strength of the applied perturbation. This perturbation is added to the input image and optionally clamped in the case of images to ensure all pixels are in $[0, 1]$. Algorithm 1 describes this process precisely.

Algorithm 1: Generating viewmaker views

```
Input: Viewmaker network $V$, $C \times W \times H$ image X, $\ell_1$ distortion budget $\epsilon$, noise $\delta$
Output: Perturbed $C \times W \times H$ image $X$
$P \leftarrow V(X, \delta)$ // generate perturbation
$P \leftarrow \frac{\epsilon CWH}{|P|_1} P$ // project to $\ell_1$ sphere
$X \leftarrow X + P$ // apply perturbation
$X \leftarrow \text{clamp}(X, 0, 1)$ // clamp (images only)
```

4 Images

We begin by applying the viewmaker to contrastive learning for images. In addition to SimCLR (Chen et al., 2020a), we also consider a memory bank-based instance discrimination framework (Wu et al., 2018, henceforth InstDisc).

We pretrain ResNet-18 (He et al., 2015) models on CIFAR-10 (Krizhevsky, 2009) for 200 epochs with a batch size of 256. We train a viewmaker-encoder system with a distortion budget of $\epsilon = 0.05$. We tried distortion budgets $\epsilon \in \{0.1, 0.05, 0.02\}$ and found 0.05 to work best; however, we anticipate that further tuning would yield additional gains. As we can see in Figure 1, the learned views are diverse, consisting of qualitatively different kinds of perturbations and affecting different parts of the input. We compare the resulting encoder representations with a model trained with the *expert views* used for SimCLR, comprised of many human-defined transformations targeting different kinds of invariances useful for image classification: cropping-and-resizing, blurring, horizontal flipping, color dropping, and shifts in brightness, contrast, saturation, and hue (Chen et al., 2020a).

4.1 Transfer results on image classification tasks

We evaluate our models on CIFAR-10, as well as eleven transfer tasks including MetaDataset (Triantafillou et al., 2019), MSCOCO (Lin et al., 2014), MNIST (LeCun et al., 1998), and FashionMNIST (Xiao et al., 2017). We use the standard linear evaluation protocol, which trains a logistic

	SimCLR		InstDisc			SimCLR		InstDisc	
Dataset	Expt	Ours	Expt	Ours	Dataset	Expt	Ours	Expt	Ours
CIFAR-10	**86.2**	84.5	**82.4**	80.1	MNIST	97.1	**98.7**	98.7	**98.9**
MSCOCO	49.9	**50.4**	48.6	**50.2**	FaMNIST	88.3	**91.5**	89.2	**91.4**
CelebA (F1)	51.0	**51.8**	**57.0**	53.7	CUBirds	**11.2**	8.7	**13.7**	9.4
LSUN	**56.2**	55.0	**56.0**	55.6	VGGFlower	53.3	**53.6**	**61.5**	54.8
Aircraft	**32.5**	31.7	**37.7**	33.5	TrafficSign	**96.6**	94.9	**98.9**	94.3
DTD	**30.4**	28.8	**29.8**	**29.8**	Fungi	**2.2**	2.0	**2.6**	2.1

Table 1: **Our learned views (Ours) enable comparable transfer performance to expert views (Expt) on CIFAR-10.** Suite of transfer tasks using pretrained representations from CIFAR-10 for both the SimCLR and InstDisc pretraining setups. Numbers are percent accuracy with the exception of CelebA which is F1. FaMNIST stands for FashionMNIST.

regression on top of representations from a frozen model. We apply the same views as in pretraining, freezing the final viewmaker when using learned views; we apply no views during validation. Table 1 shows our results, indicating comparable overall performance with SimCLR and InstDisc, all without the use of human-crafted view functions. This performance is noteworthy as our ℓ_1 views cannot implement cropping-and-rescaling, which was shown to be the most important view function in Chen et al. (2020a). We speculate that the ability of the viewmaker to implement partial masking of an image may enable a similar kind of spatial information ablation as cropping.

4.1.1 Comparison to random ℓ_1 noise

Is random noise sufficient to produce domain-agnostic views? To assess how important adversarial training is to the quality of the learned representations, we perform an ablation where we generate views by adding Gaussian noise normalized to the same $\epsilon = 0.05$ budget as used in the previous section. Transfer accuracy on CIFAR-10 is significantly hurt by this ablation, reaching **52.01%** for a SimCLR model trained with random noise views compared to **84.50%** for our method, demonstrating the importance of adversarial training to our method.

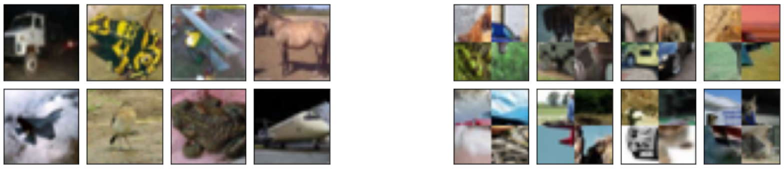
4.1.2 The importance of inter-patch mutual information and cropping views

Cropping-and-resizing has been identified as a crucial view function when pretraining on ImageNet (Chen et al., 2020a). However, what properties of a pretraining dataset make cropping useful? We hypothesize that such a dataset must have images whose patches have high mutual information. In other words, there must be some way for the model to identify that different patches of the same image come from the same image. While this may be true for many object or scene recognition datasets, it may be false for other important pretraining datasets, including medical or satellite imagery, where features of interest are isolated to particular parts of the image.

To investigate this hypothesis, we modify the CIFAR-10 dataset to reduce the inter-patch mutual information by replacing each 16x16 corner of the image with the corner from another image in the training dataset (see Figure 3 for an example). Thus, random crops on this dataset, which we call CIFAR-10-Corners, will often contain completely unrelated information. When pretrained on CIFAR-10-Corners, expert views achieve **63.3%** linear evaluation accuracy on the original CIFAR-10 dataset, while viewmaker views achieve **68.8%**. This gap suggests that viewmaker views are less reliant on inter-patch mutual information than the expert views.

4.2 Combining viewmaker and handcrafted views

Can viewmakers improve performance in cases where some useful handcrafted views have already been identified? Chen et al. (2020a) show that views produced through cropping are significantly improved by a suite of color-based augmentations, which they argue prevents the network from relying solely on color statistics to perform the contrastive task. Here, we show that viewmaker networks also enable strong gains when added on top of cropping and horizontal flipping views when pretraining on CIFAR-10—without any domain-specific knowledge. Alone, this subset of



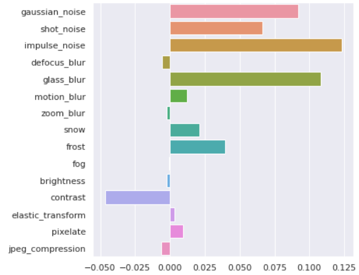
(a) CIFAR-10

(b) CIFAR-10-Corners

Figure 3: Our learned views are still able to yield useful information even when the inter-patch mutual information in a dataset is low, as in Figure 3b.

Views	Clean	Corrupted	Diff
Ours	84.5	71.4	-13.1
SimCLR*	**86.2**	77.1	-9.1
Combined*	**86.3**	**79.8**	**-6.5**

(a) Accuracy on CIFAR-10 and CIFAR-10-C. *Overlap with CIFAR-10-C corruptions.



(b) Accuracy gain on CIFAR-10-C by from adding our learned views atop expert views.

Figure 4: **Performance of different views on CIFAR-10-C corruptions.** Our learned views enable solid performance in the face of unseen corruptions despite not explicitly including any blurring, contrast, or brightness transformations during training, unlike the expert views. Adding our learned views on top of SimCLR yields additional gains in robust accuracy, especially on different kinds of noise corruptions and glass blurring.

handcrafted augmentations achieves **73.2%** linear evaluation accuracy on CIFAR-10. Combining these views with learned viewmaker perturbations ($\epsilon = 0.05$) achieves **83.1%**.[1] This suggests that viewmakers can significantly improve representation learning even in cases where some domain-specific views have already been developed.

4.3 Robustness to common corruptions

Image classification systems should behave robustly even when the data distribution is slightly different from that seen during training. Does using a viewmaker improve robustness against common types of corruptions not experienced at train time? To answer this, we evaluate both learned views, expert views, and their composition on the CIFAR-10-C dataset (Hendrycks & Dietterich, 2019), which assesses robustness to corruptions like snow, pixelation, and blurring. In this setting, corruptions are applied only at test time, evaluating whether the classification system is robust to some types of corruptions to which humans are robust.

When considering methods in isolation, SimCLR augmentations result in less of an accuracy drop from clean to corrupted data compared to our learned views, as shown in Table 4a. This gap is expected, as the expert views overlap significantly with the CIFAR-10-C corruptions: both include blurring, brightness, and contrast transformations. Interestingly, however, when we train a viewmaker network while also applying expert augmentations ("Combined," Table 4a), we can further improve the robust accuracy, with notable gains on noise and glass blur corruptions (Figure 4b). This is noteworthy, as our learned views have no explicit overlap with the CIFAR-10-C corruptions, unlike the expert augmentations.[2] In the Combined setting, we use a distortion budget of $\epsilon = 0.01$,

[1] We did not see additional gains from using viewmakers on top of the full, well-optimized set of SimCLR augmentations.

[2] We do notice a smaller decline in contrast corruption accuracy, possibly due to interactions between changing pixel magnitudes and the ℓ_p constraint.

	Expert		Ours (ϵ)	
ResNet-18, 100hr	Time	Spec.	0.05	0.1
LibriSpeech Sp. ID	**97.1**	91.6	88.3	84.0
VoxCeleb1 Sp. ID	5.7	7.8	**12.1**	9.1
AudioMNIST	31.7	63.9	**93.3**	87.9
Google Commands	27.1	31.9	**47.4**	41.6
Fluent Actions	29.4	32.0	**41.6**	37.9
Fluent Objects	37.1	40.3	**47.6**	**47.6**
Fluent Locations	59.7	63.3	66.5	**68.3**

ResNet-50, 960hr	Spec.	0.05
LibriSpeech Sp. ID	**95.9**	90.0
VoxCeleb1 Sp. ID	8.6	**10.7**
AudioMNIST	80.2	**88.0**
Google Commands	28.3	**32.6**
Fluent Actions	30.5	**42.5**
Fluent Objects	36.2	**50.8**
Fluent Locations	62.0	**68.9**

Table 2: **Our learned views significantly outperform existing views for speech transfer tasks.** Linear evaluation accuracy for SimCLR models trained on LibriSpeech. Left: ResNet-18 + Librispeech 100 hour, Right: ResNet-50 + Librispeech 960hr. "Time" refers to view functions applied in the time domain (Kharitonov et al., 2020), while "Spec." refers to view functions applied directly to the spectrogram (Park et al., 2019). 0.05 and 0.1 denote viewmaker distortion bounds ϵ.

which we find works better than $\epsilon = 0.05$, likely because combining the two augmentations at their full strength would make the learning task too difficult.

These results suggest that learned views are a promising avenue for improving robustness in self-supervised learning models.

5 Speech

Representation learning on speech data is an emerging and important research area, given the large amount of available unlabeled data and the increasing prevalence of speech-based human-computer interaction (Latif et al., 2020). However, compared to images, there is considerably less work on self-supervised learning and data augmentations for speech data. Thus, it is a compelling setting to investigate whether viewmaker augmentations are broadly applicable across modalities.

5.1 Self-supervised learning setup

We adapt the contrastive learning setup from SimCLR (Chen et al., 2020a). Training proceeds largely the same as for images, but the inputs are 2D log mel spectrograms. We consider both view functions applied in the time-domain before the STFT, including noise, reverb, pitch shifts, and changes in loudness (Kharitonov et al., 2020), as well as spectral views, which involve masking or noising different parts of the spectrogram (Park et al., 2019). To generate learned views, we pass the spectrogram as input to the viewmaker. We normalize the spectrogram to mean zero and variance one before passing it through the viewmaker, and do not clamp the resulting perturbed spectrogram. See the Appendix for more details. We train on the Librispeech dataset (Panayotov et al., 2015) for 200 epochs, and display some examples of learned views in the Appendix.

5.2 Speech classification results

We evaluate on three speech classification datasets: Fluent Speech Commands (Lugosch et al., 2019), Google Speech Commands (Warden, 2018), and spoken digit classification (Becker et al., 2018), as well as speaker classification on VoxCeleb (Nagrani et al., 2017) and Librispeech (Panayotov et al., 2015), all using the linear evaluation protocol for 100 epochs. In Table 2, we report results with both the same distortion budget $\epsilon = 0.05$ as in the image domain, as well as a larger $\epsilon = 0.1$, for comparison. Both versions significantly outperform the preexisting waveform and spectral augmentations, with a $+9$ percentage point improvement on average for the ResNet-18 ($\epsilon = 0.05$) viewmaker model over the best expert views. The gains for real-world tasks such as command identification are compelling. One notable exception is the task of LibriSpeech speaker identification. Since LibriSpeech is the same dataset the model was pretrained on, and this effect is not replicated on VoxCeleb1, the other speaker classification dataset, we suspect the model may be picking up on dataset-specific artifacts (e.g. background noise, microphone type) which may make the speaker

	Spectral		Ours (ϵ)				
Dataset	With Noise	Without Noise	0.02	0.05	0.2	0.5	2.0
Pamap2	71.0	74.6	83.0	87.4	88.6	**91.3**	9.1

Table 3: **Our learned views significantly outperform existing views for activity recognition on wearable sensor data.** Our method learns superior representations across a large range of distortion budgets ϵ, although budgets that are too strong prevent learning. Linear evaluation accuracy for ResNet18 models trained on Pamap2 with SimCLR. "Spectral" refers to view functions applied directly to the spectrogram (Park et al., 2019).

ID task artificially easier. An interesting possibility is that the worse performance of viewmaker views may result from the model being able to identify and ablate such spurious correlations in the spectrograms.

6 WEARABLE SENSOR DATA

To further validate that our method for learning views is useful across different modalities, we consider time-series data from wearable sensors. Wearable sensor data has a broad range of applications, including health care, entertainment, and education (Lara & Labrador, 2012). We specifically consider whether viewmaker views improve representation learning for the task of human activity recognition (HAR), for example identifying whether a user is jumping rope, running, or cycling.

6.1 SELF-SUPERVISED LEARNING SETUP

We consider the Pamap2 dataset (Reiss & Stricker, 2012), a dataset of 12 different activities performed by 9 participants. Each activity contains 52 different time series, including heart rate, accelerometer, gyroscope, and magnetometer data collected from sensors on the ankle, hand, and chest (all sampled at 100Hz, except heart rate, which is sampled at approximately 9Hz). We linearly interpolate missing data, then take random 10s windows from subject recordings, using the same train/validation/test splits as prior work (Moya Rueda et al., 2018). To create inputs for our model, we generate a multi-channel image composed of one 32x32 log spectrogram for each sensor time-series window. Unlike speech data, we do not use the mel scale when generating the spectrogram. We then normalize the training and validation datasets by subtracting the mean and then dividing by the standard deviation of the training dataset.

We train with both our learned views and the spectral views (Park et al., 2019) that were most successful in the speech domain (for multi-channel spectral masking, we apply the same randomly chosen mask to all channels). We also compare against a variant of these views with spectrogram noise removed, which we find improves this baseline's performance.

6.2 SENSOR-BASED ACTIVITY RECOGNITION RESULTS

We train a linear classifier on the frozen encoder representations for 50 epochs, reporting accuracy on the validation set. We sample 10k examples for each training epoch and 50k examples for validation. Our views significantly outperform spectral masking by 12.8 percentage points when using the same $\epsilon = 0.05$ as image and speech, and by 16.7 points when using a larger $\epsilon = 0.5$ (Table 3). We also find that a broad range of distortion budgets produces useful representations, although overly-aggressive budgets prevent learning (Table 3). These results provide further evidence that our method for learning views has broad applicability across different domains.

6.3 SEMI-SUPERVISED EXPERIMENTS

An especially important setting for self-supervised learning is domains where labeled data is scarce or costly to acquire. Here, we show that our method can enable strong performance when labels for only a single participant (Participant 1) out of seven are available. We compare simple supervised learning on Participant 1's labels against linear evaluation of our best pretrained model, which was trained on unlabeled data from all 7 participants. The model architectures and training procedures are otherwise identical to the previous section. As Figure 4 shows, pretraining with our method on unlabeled data enables significant gains over pure supervised learning when data is scarce, and even slightly outperforms the hand-crafted views trained on all 7 participants (cf. Table 3).

	Supervised Learning		Pretrain (Ours) & Transfer	
Dataset	1 Participant	7 Participants	1 Participant	7 Participants
Pamap2	58.3	97.1	75.1	91.3

Table 4: **Our method enables superior results in a semi-supervised setting where labels for data from only one participant are available.** Validation accuracy for activity recognition on Pamap2. Supervised Learning refers to training a randomly initialized model on the labeled data until convergence. Pretrain & Transfer refers to training a linear classifier off of the best pretrained model above. 1 or 7 Participants refers to the number of participants comprising the training set.

7 Conclusion

We introduce a method for learning views for unsupervised learning, demonstrating its effectiveness through strong performance on image, speech, and wearable sensor modalities. Our novel generative model—viewmaker networks—enables us to efficiently learn views as *part of* the representation learning process, as opposed to relying on domain-specific knowledge or costly trial and error. There are many interesting avenues for future work. For example, while the ℓ_1 constraint is simple by design, there may be other kinds of constraints that enable richer spaces of views and better performance. In addition, viewmaker networks may find use in supervised learning, for the purposes of data augmentation or improving robustness. Finally, it is interesting to consider what happens as the viewmaker networks increase in size: do we see performance gains or robustness-accuracy trade-offs (Raghunathan et al., 2019)? Ultimately, our work is a step towards more general self-supervised algorithms capable of pretraining on arbitrary data and domains.

Acknowledgements

We would like to thank Dan Yamins, Chengxu Zhuang, Shyamal Buch, Jesse Mu, Jared Davis, Aditi Raghunathan, Pranav Rajpurkar, Margalit Glasgow, and Jesse Michel for useful discussions and comments on drafts. AT is supported by an Open Phil AI Fellowship. MW is supported by the Stanford Interdisciplinary Graduate Fellowship as the Karr Family Fellow.

References

Nasir Ahmed, T_ Natarajan, and Kamisetty R Rao. Discrete cosine transform. *IEEE transactions on Computers*, 100(1):90–93, 1974.

Antreas Antoniou, Amos Storkey, and Harrison Edwards. Data augmentation generative adversarial networks, 2017.

Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *Advances in Neural Information Processing Systems*, pp. 15535–15545, 2019.

Suzanna Becker and Geoffrey E Hinton. Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, 355(6356):161–163, 1992.

Sören Becker, Marcel Ackermann, Sebastian Lapuschkin, Klaus-Robert Müller, and Wojciech Samek. Interpreting and explaining deep neural networks for classification of audio signals, 2018.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Lukas Biewald. Experiment tracking with weights and biases, 2020. URL `https://www.wandb.com/`. Software available from wandb.com.

Avishek Joey Bose, Huan Ling, and Yanshuai Cao. Adversarial contrastive estimation. *arXiv preprint arXiv:1805.03642*, 2018.

Christopher Bowles, Liang Chen, Ricardo Guerrero, Paul Bentley, Roger Gunn, Alexander Hammers, David Alexander Dickie, Maria Valdés Hernández, Joanna Wardlaw, and Daniel Rueckert. Gan augmentation: Augmenting training data using generative adversarial networks, 2018.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020.

Olivier Chapelle, Jason Weston, Léon Bottou, and Vladimir Vapnik. Vicinal risk minimization. In *Advances in neural information processing systems*, pp. 416–422, 2001.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020a.

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020b.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020c.

Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018.

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 702–703, 2020.

Terrance DeVries and Graham W. Taylor. Dataset augmentation in feature space, 2017.

Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning. In *Advances in Neural Information Processing Systems*, pp. 10542–10552, 2019.

Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016.

Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *Advances in neural information processing systems*, pp. 766–774, 2014.

WA Falcon. Pytorch lightning. *GitHub. Note: https://github.com/PyTorchLightning/pytorch-lightning*, 3, 2019.

Raphael Gontijo-Lopes, Sylvia J Smullin, Ekin D Cubuk, and Ethan Dyer. Affinity and diversity: Quantifying mechanisms of data augmentation. *arXiv preprint arXiv:2002.08973*, 2020.

Ian Goodfellow. Nips 2016 tutorial: Generative adversarial networks. *arXiv preprint arXiv:1701.00160*, 2016.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.

Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pp. 1735–1742. IEEE, 2006.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. *arXiv preprint arXiv:2006.16241*, 2020.

Daniel Ho, Eric Liang, Xi Chen, Ion Stoica, and Pieter Abbeel. Population based augmentation: Efficient learning of augmentation policy schedules. In *International Conference on Machine Learning*, pp. 2731–2741. PMLR, 2019.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (eds.), *Computer Vision – ECCV 2016*, pp. 694–711, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46475-6.

Eugene Kharitonov, Morgane Rivière, Gabriel Synnaeve, Lior Wolf, Pierre-Emmanuel Mazaré, Matthijs Douze, and Emmanuel Dupoux. Data augmenting contrastive learning of speech representations in the time domain. *arXiv preprint arXiv:2007.00991*, 2020.

Minseon Kim, Jihoon Tack, and Sung Ju Hwang. Adversarial self-supervised contrastive learning, 2020.

György Kovács, László Tóth, Dirk Van Compernolle, and Sriram Ganapathy. Increasing the robustness of cnn acoustic models using autoregressive moving average spectrogram features and channel dropout. *Pattern Recognition Letters*, 100:44–50, 2017.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

Oscar D Lara and Miguel A Labrador. A survey on human activity recognition using wearable sensors. *IEEE communications surveys & tutorials*, 15(3):1192–1209, 2012.

Siddique Latif, Rajib Rana, Sara Khalifa, Raja Jurdak, Junaid Qadir, and Björn W Schuller. Deep representation learning in speech processing: Challenges, recent advances, and future trends. *arXiv preprint arXiv:2001.00378*, 2020.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Sungbin Lim, Ildoo Kim, Taesup Kim, Chiheon Kim, and Sungwoong Kim. Fast autoaugment. In *Advances in Neural Information Processing Systems*, pp. 6665–6675, 2019.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Raphael Gontijo Lopes, Dong Yin, Ben Poole, Justin Gilmer, and Ekin D Cubuk. Improving robustness without sacrificing accuracy with patch gaussian augmentation. *arXiv preprint arXiv:1906.02611*, 2019.

Loren Lugosch, Mirco Ravanelli, Patrick Ignoto, Vikrant Singh Tomar, and Yoshua Bengio. Speech model pre-training for end-to-end spoken language understanding, 2019.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Matthias Minderer, Olivier Bachem, Neil Houlsby, and Michael Tschannen. Automatic shortcut removal for self-supervised representation learning. *arXiv preprint arXiv:2002.08822*, 2020.

Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6707–6717, 2020.

Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993, 2018.

Fernando Moya Rueda, René Grzeszick, Gernot A Fink, Sascha Feldhorst, and Michael Ten Hompel. Convolutional neural networks for human activity recognition using body-worn sensors. In *Informatics*, volume 5, pp. 26. Multidisciplinary Digital Publishing Institute, 2018.

Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. Voxceleb: a large-scale speaker identification dataset, 2017.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An asr corpus based on public domain audio books. pp. 5206–5210, 04 2015. doi: 10.1109/ICASSP.2015.7178964.

Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*, 2019.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pp. 8026–8037, 2019.

Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017.

Senthil Purushwalkam and Abhinav Gupta. Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases. *arXiv preprint arXiv:2007.13916*, 2020.

Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. Certified defenses against adversarial examples. *arXiv preprint arXiv:1801.09344*, 2018.

Aditi Raghunathan, Sang Michael Xie, Fanny Yang, John C Duchi, and Percy Liang. Adversarial training can hurt generalization. *arXiv preprint arXiv:1906.06032*, 2019.

Alexander J Ratner, Henry Ehrenberg, Zeshan Hussain, Jared Dunnmon, and Christopher Ré. Learning to compose domain-specific transformations for data augmentation. In *Advances in neural information processing systems*, pp. 3236–3246, 2017.

Attila Reiss and Didier Stricker. Introducing a new benchmarked dataset for activity monitoring. In *2012 16th International Symposium on Wearable Computers*, pp. 108–109. IEEE, 2012.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Mehdi SM Sajjadi, Giambattista Parascandolo, Arash Mehrjou, and Bernhard Schölkopf. Tempered adversarial networks. *arXiv preprint arXiv:1802.04374*, 2018.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. *arXiv preprint arXiv:2005.10243*, 2020.

László Tóth, György Kovács, and Dirk Van Compernolle. A perceptually inspired data augmentation method for noise robust cnn acoustic models. In *International Conference on Speech and Computer*, pp. 697–706. Springer, 2018.

Toan Tran, Trung Pham, Gustavo Carneiro, Lyle Palmer, and Ian Reid. A bayesian data augmentation approach for learning deep models, 2017.

Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, and Hugo Larochelle. Meta-dataset: A dataset of datasets for learning to learn from few examples, 2019.

Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation, 2018.

Pete Warden. Speech commands: A dataset for limited-vocabulary speech recognition, 2018.

Eric Wong and J. Zico Kolter. Learning perturbation sets for robust machine learning, 2020.

Mike Wu, Chengxu Zhuang, Milan Mosse, Daniel Yamins, and Noah Goodman. On mutual information in contrastive learning for visual representations. *arXiv preprint arXiv:2005.13149*, 2020.

Zhirong Wu, Yuanjun Xiong, Stella Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance-level discrimination. *arXiv preprint arXiv:1805.01978*, 2018.

Chaowei Xiao, Bo Li, Jun-Yan Zhu, Warren He, Mingyan Liu, and Dawn Song. Generating adversarial examples with adversarial networks, 2018.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 6023–6032, 2019.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Xinyu Zhang, Qiang Wang, Jian Zhang, and Zhao Zhong. Adversarial autoaugment. *arXiv preprint arXiv:1912.11188*, 2019.

Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 6002–6012, 2019.

A ADDITIONAL EXPERIMENTAL DETAILS

A.1 IMAGE EXPERIMENTS

The primary image experiments compare SimCLR and the instance discrimination method from (Wu et al., 2018) (henceforth InstDisc) with and without the viewmaker on a suite of transfer datasets.

For pretraining, we use a ResNet-18 encoder without the maxpool layer after the first convolutional layer. We found removing this to be crucial for performance across all models when working with smaller input images of 32x32 pixels. We use an embedding dimension of size 128 and do not use an additional projection head as in Chen et al. (2020b). For the SimCLR objective, we use a temperature of 0.07. For the InstDisc objective, we use 4096 negative samples from the memory bank and an update rate of 0.5. We optimize using SGD with batch size 256, learning rate 0.03, momentum 0.9, and weight decay 1e-4 for 200 epochs with no learning rate dropping, which we found to hurt performance in CIFAR-10.

For the viewmaker, we adapt the style transfer network from Johnson et al. (2016), using a PyTorch implementation,[3] but use three residual blocks instead of five, which we found did not hurt performance despite the reduced computation. To add stochasticity, we concatenate a uniform random noise channel to the input and the activations before each residual block.

Additionally, we performed preliminary experiments with a U-Net architecture (Ronneberger et al., 2015) for the viewmaker but found significantly worse performance. We leave a more in-depth investigation of the role of architecture and model size in the effectiveness of the viewmaker.

During transfer (linear evaluation), we use the pre-pooling features after the last convolutional layer of the ResNet-18, totaling 512*7*7 dimensions. We load the parameters from the final iteration of pretraining. We optimize a logistic regression model with the frozen ResNet-18 model using SGD with learning rate 0.01, momentum 0.9, weight decay 0, batch size 128 for 100 epochs. We drop the learning rate by a factor of 10 on epochs 60 and 80. We preprocess the training and validation datasets by subtracting and dividing by the mean and standard deviation of the training dataset, respectively. For models trained with a viewmaker network, we load and freeze the final viewmaker checkpoint to supply augmentations during transfer training. Otherwise, we use the same expert views used during pretraining.

The CIFAR-10-Corners experiments were conducted in the same way, except that the transfer task is the original CIFAR-10 dataset.

For the robustness experiments on CIFAR-10-C, the final transfer model trained on CIFAR-10 was evaluated without further training on the CIFAR-10-C dataset.

A.2 SPEECH EXPERIMENTS

The setup for the speech experiments is almost identical to images. The primary distinction is in preprocessing the data. In our experiments, pretraining is done on two splits of LibriSpeech: a smaller set containing 100 hours of audio and a larger set containing 960 hours of audio. Each instance is a raw waveform. We pick a maximum limit of 150k frames and truncate waveforms containing more frames. We randomly pick whether to truncate the beginning or end of the waveform during training, whereas for evaluation, we always truncate from the end. Next, we compute log mel spectrograms on the truncated waveforms as the input to our encoder. For 100hr LibriSpeech, we use a hop length of 2360 and set the FFT window length to be 64, resulting in a 64x64 tensor. For the 960hr LibriSpeech, we wanted to show our method generalizes to larger inputs, so we use a hop length of 672 with an FFT window length of 112 for a tensor of size 112x112. Finally, we log the spectrogram by squaring it and converting power to decibels.

For expert views, we consider both a method that applies views directly to the waveforms (Kharitonov et al., 2020) and a method that does so on the resulting spectrograms (Park et al., 2019). For the former, we use code from the NLPAUG library[4] to take a random contiguous crop of the

[3] https://github.com/pytorch/examples/tree/master/fast_neural_style

[4] https://github.com/makcedward/nlpaug

waveform with scale (0.08,1.0) and add Gaussian noise with scale 1. We randomly mask contiguous segments on the horizontal (frequency) and vertical (time) axes for the latter.

To do this, we also use the NLPAUG library and employ the FREQUENCYMASKINGAUG and TIMEMASKINGAUG functions with MASK_FACTOR set to 40. Having done this, we are left with a 1x64x64 tensor for the 100-hour dataset or a 1x112x112 tensor for the 960-hour dataset. For the former, we use the same ResNet-18 as described above; pretraining and transfer use the same hyperparameters. In the latter, we use a ResNet-50 encoder with an MLP projection head with a hidden dimension of 2048. We use TORCHVISION implementations (Paszke et al., 2019). We still use the pre-pooling features for transfer in this setting as we found better performance than using post-pooling features. Otherwise, hyperparameters are identical to the 100-hour setting (and the CIFAR-10 setting).

For each transfer dataset, we convert waveforms to normalized spectrograms in the same manner as just described. The AudioMNIST dataset was downloaded from `https://github.com/soerenab/AudioMNIST`; Google Speech Commands was downloaded from `https://ai.googleblog.com/2017/08/launching-speech-commands-dataset.html`; Fluent Speech Commands was downloaded from `https://fluent.ai/fluent-speech-commands-a-dataset-for-spoken-language-understanding-research`; VoxCeleb1 was downloaded from `http://www.robots.ox.ac.uk/~vgg/data/voxceleb/vox1.html` (we use version 1 of the corpus). Each transfer dataset was again normalized using the training split's mean and standard deviation.

A.3 WEARABLE SENSOR EXPERIMENTS

The experimental paradigm for wearable sensor data largely follows that for speech. To generate an example, we randomly sample a subject (from the correct training split) and activity; we next randomly sample a contiguous 10s frame, linearly interpolating missing data. We generate spectrograms for each of the 52 sensors without Mel scaling, using 63 FFT bins, a hop length of 32, and a power of 2, then take the logarithm after adding 1e-6 for numerical stability. This process yields 52 32x32 spectrograms, which we treat as different channels, giving a tensor of shape [52, 32, 32]. We then normalize the spectrograms by subtracting and dividing by the mean and standard deviation of 10k samples from the training set.

A.4 TRAINING COSTS

We train all models on single NVIDIA Titan XP GPUs. On our system, training with a viewmaker network roughly increased training time by 50% and GPU memory utilization by 100%.

A.5 FRAMEWORKS

We make use of PyTorch (Paszke et al., 2019), PyTorch Lightning (Falcon, 2019), and Weights & Biases (Biewald, 2020) for our experiments.

B ADDITIONAL GENERATED VIEWS

B.1 CIFAR-10 VIEWS

We visualize more views for CIFAR in Figure 5. We also visualize the difference between examples and their views (rescaled to [0,1]) in Figure 6. These figures further demonstrate the complexity, diversity, and input dependence of the viewmaker views.

B.1.1 APPLYING PERTURBATION IN THE FREQUENCY DOMAIN

Are there other natural ways of generating perturbations with bounded complexity? One other technique we considered was applying views in the frequency domain. Specifically, we apply a Discrete Cosine Transform (Ahmed et al., 1974, DCT) before applying the ℓ_1-bounded perturbation, then

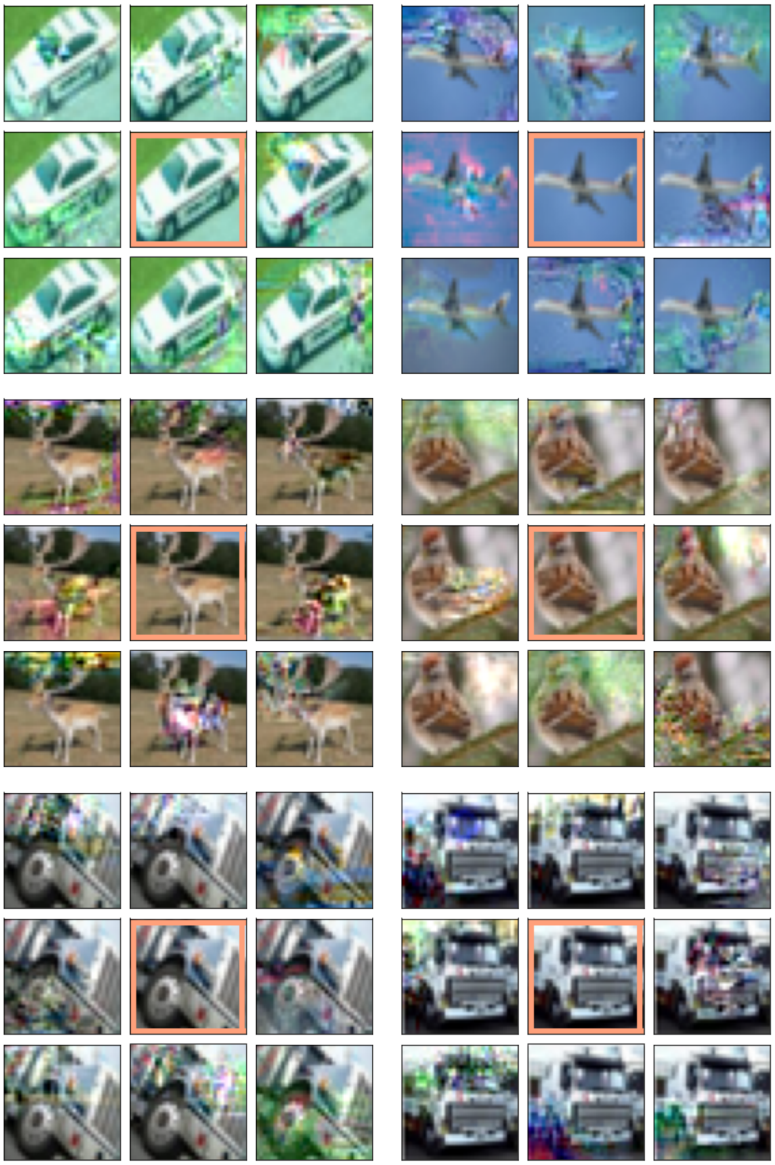


Figure 5: Learned views for random CIFAR-10 examples. Original image shown in center, with pink border. Distortion budget is $\epsilon = 0.05$.

apply the inverse DCT to obtain an image in the original domain. We use a PyTorch library[5] to compute the DCT, which is a differentiable transform. After a coarse hyperparameter search, we achieved the best results with $\epsilon = 1.0$: **74.4%** linear evaluation accuracy on CIFAR-10, much lower than our other models. However, the views are still illustrative, and we show some examples in Figure 7.

B.2 LIBRISPEECH VIEWS

We visualize some views for random LibriSpeech spectrograms in Figure 8, as well as showing deltas between spectrograms and views in Figure 9. The figures show how the viewmaker applies a variety of kinds of perturbations across the entire spectrogram.

[5] `https://github.com/zh217/torch-dct`

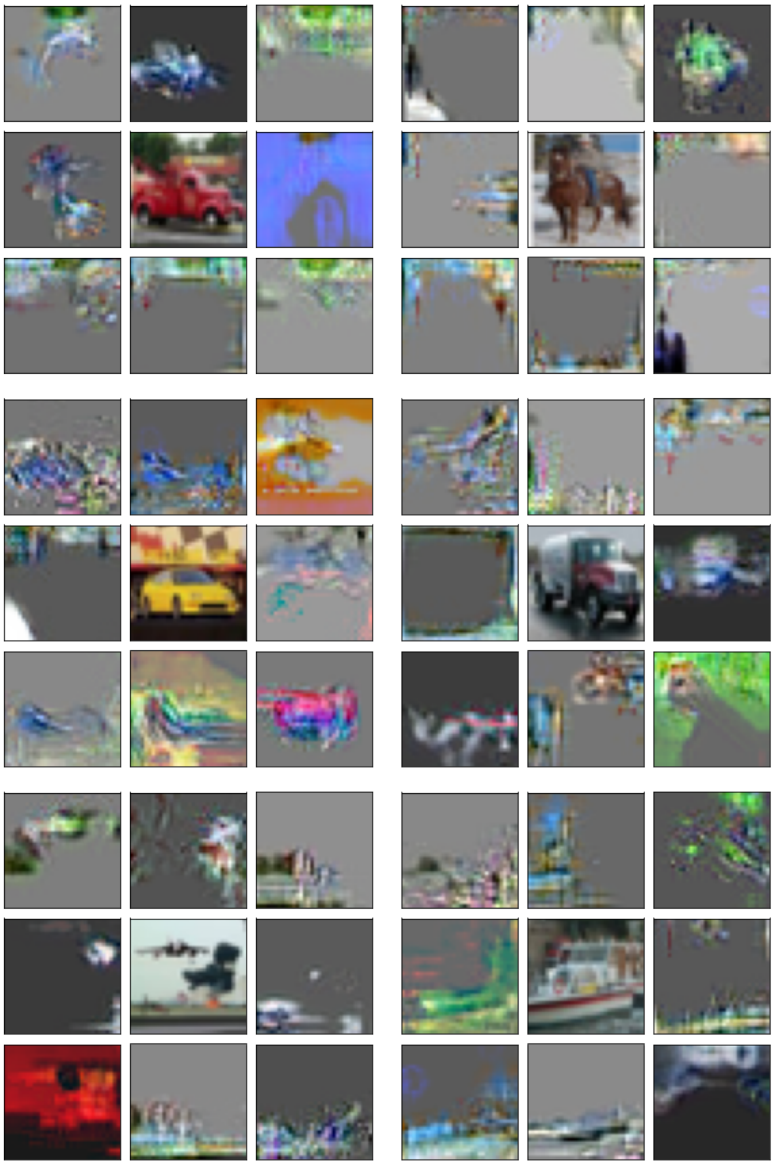


Figure 6: Difference between random CIFAR-10 examples and their viewmaker views. Original image shown in center, diffs shown on perimeter. Diffs linearly rescaled to [0, 1]. Distortion budget is $\epsilon = 0.05$.

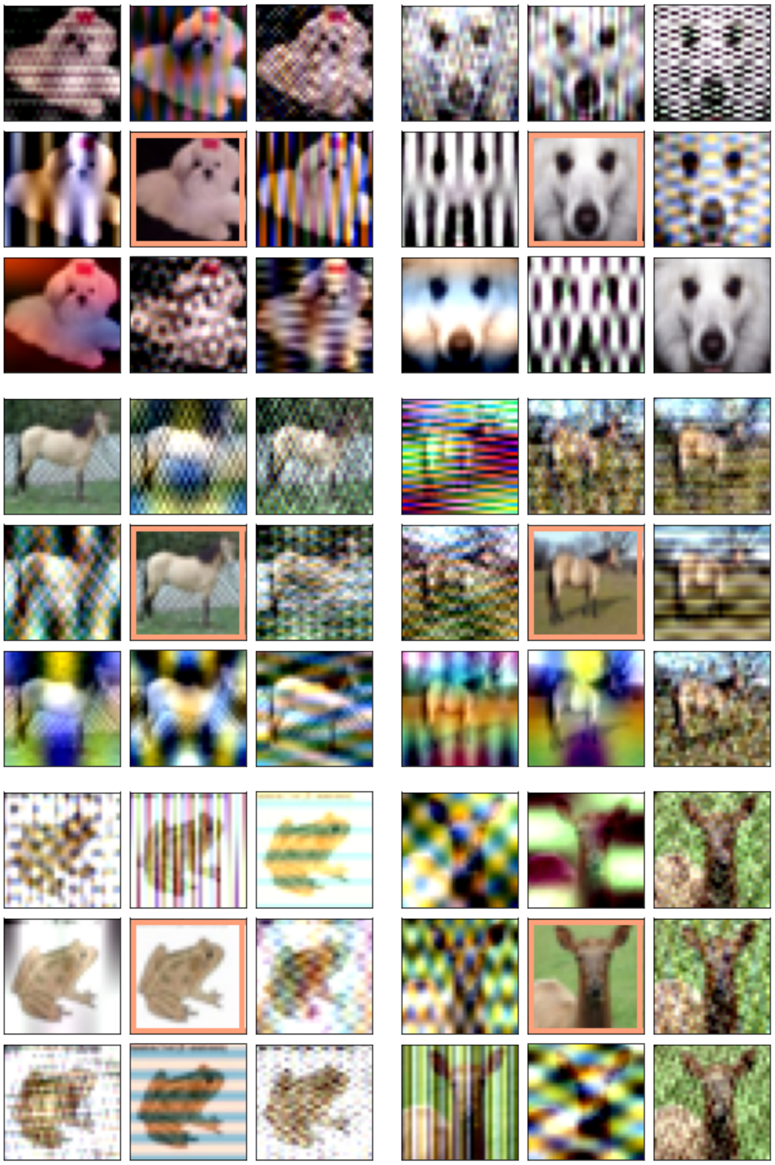


Figure 7: Learned views for random CIFAR-10 examples with perturbation applied in frequency domain. Original image shown in center, with pink border. Distortion budget is $\epsilon = 1.0$.

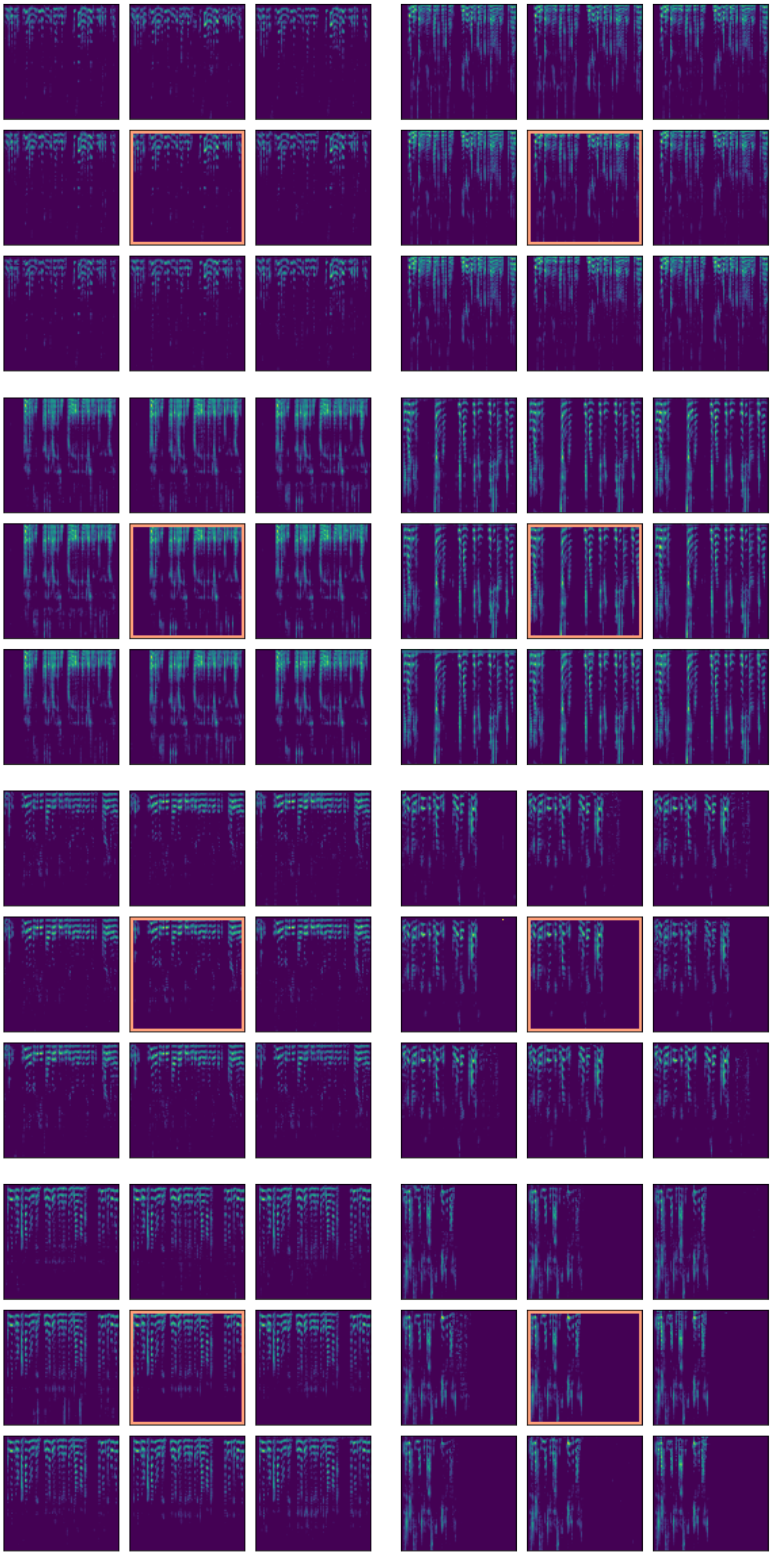


Figure 8: Examples of learned views for random Librispeech spectrograms. Original image shown in center, with pink border. Variations are subtle—best viewed at high magnification. Color scale endpoints set to minimum and maximum of original image. Spectrograms are 64x64 log mel spectrograms from LibriSpeech 100 hours. Distortion budget is $\epsilon = 0.05$.

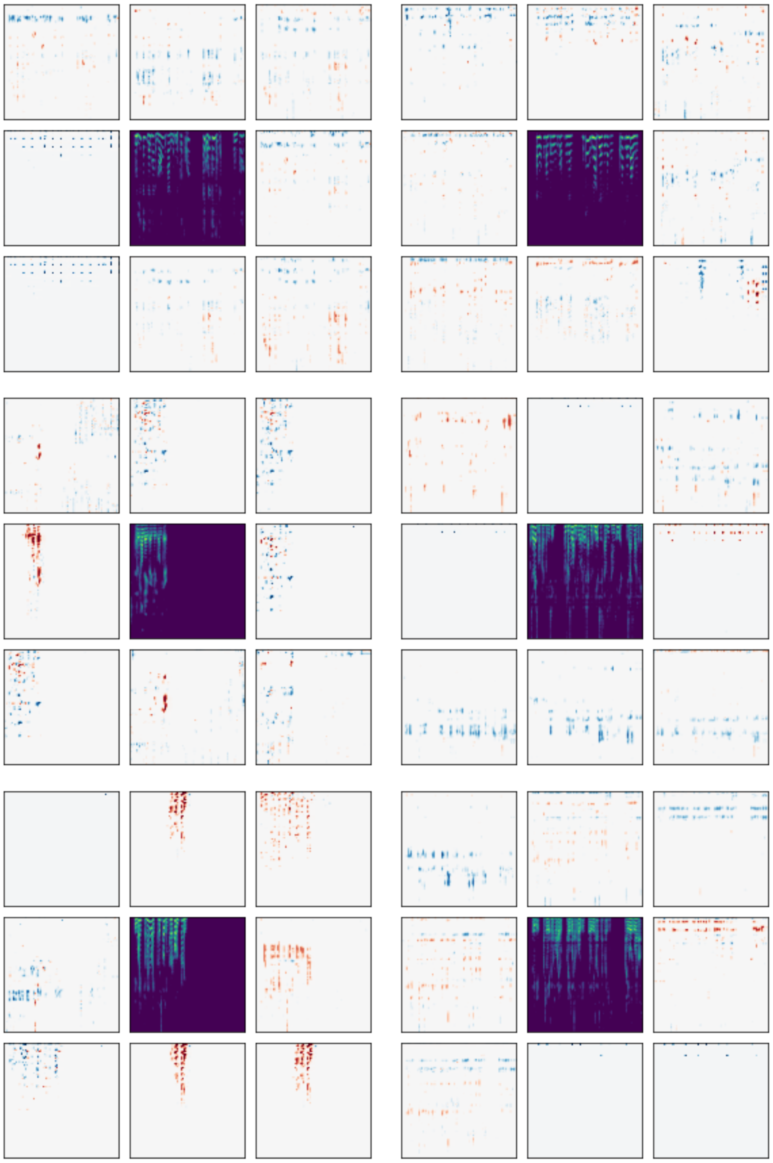


Figure 9: Difference between random LibriSpeech spectrograms and their viewmaker views. Original spectrogram shown in center, diffs shown on perimeter. Color scale endpoints set to -2.5 (red) to +2.5 (blue), although some values exceed these endpoints. Spectrograms are 64x64 log mel spectrograms from LibriSpeech 960 hours. Distortion budget is $\epsilon = 0.05$.

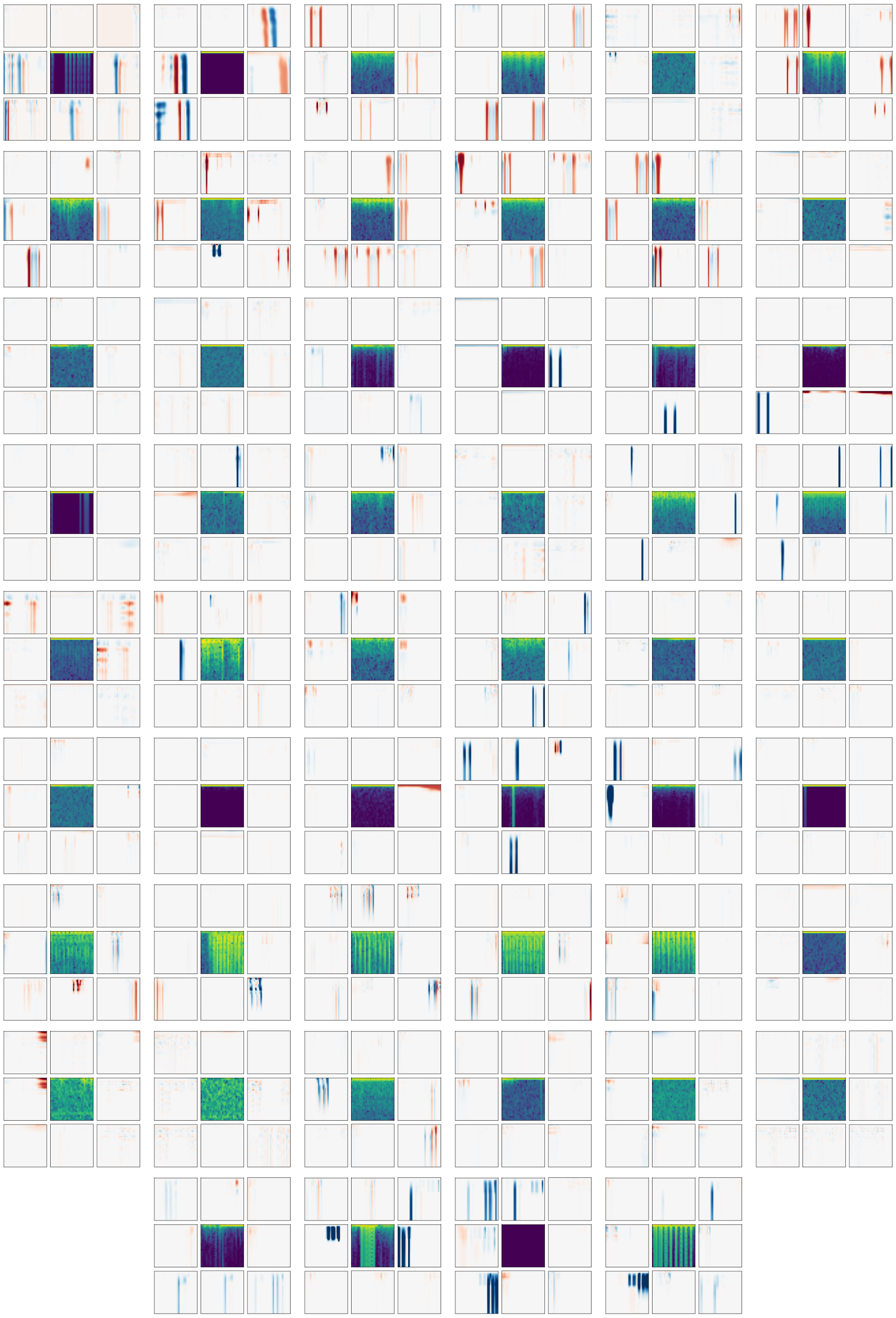


Figure 10: Difference between random Pamap2 spectrograms and their viewmaker views. Original spectrogram shown in center, diffs shown on perimeter. Each 3x3 panel shows data from a different example and sensor. Color scale endpoints set to -2 (red) to +2 (blue), although some values exceed these endpoints. Distortion budget is $\epsilon = 0.05$.

B.3 Wearable sensor views

We visualize deltas between Pamap2 spectrograms and their views in Figure 10. Each 3x3 panel shows data and views from a different sensor and example.

Dataset	Ours	Expert
Aircraft	32.0 (0.7)	32.0 (0.6)
Birds	8.7 (0.3)	10.9 (0.3)
DTD	27.8 (0.9)	30.4 (1.1)
FaMNIST	91.0 (0.4)	88.5 (0.2)
MNIST	98.8 (0.1)	97.1 (0.0)
Traffic	94.8 (1.0)	96.7 (0.3)
Flower	50.6 (2.6)	53.2 (0.4)

Table 5: **Stability of viewmaker networks across random seeds.** Linear evaluation accuracy and standard deviation for three random seeds, where the seed varies across both pretraining and transfer. Experimental setup is identical to that of Table 1.

	Expert		Ours (ϵ)	
ResNet-18, 100hr	Time	Spec.	0.05	0.1
Top-1 Accuracy	$\mathbf{97.1}$	91.6	88.3	84.0
Top-5 Accuracy	5.7	7.8	$\mathbf{12.1}$	9.1

ResNet-50, 960hr	Spec.	0.05
Top-1 Accuracy	$\mathbf{97.1}$	91.6
Top-5 Accuracy	5.7	7.8

Table 6: VoxCeleb speaker identification linear evaluation accuracy. Experimental setup identical to Table 2.

B.4 Stability across random seeds

While instability has been reported as a common issue when training GANs (Goodfellow, 2016), we encountered few optimization difficulties training viewmakers. To empirically demonstrate the stability of our approach across random seeds, we report the average and standard deviation of transfer accuracy across three pretraining and transfer runs for different datasets. The experimental setup is identical to the results presented in Table 1, and the random seeds vary across both pretraining and transfer. Table 5 shows that the observed standard deviations are small, lying within a percentage point in all but one case.

B.5 Top-5 accuracy for VoxCeleb Speaker Identification

We also present Top-5 accuracies for VoxCeleb speaker identification in Table 6, along with with Top-1 accuracies for comparison.