Image-to-Image Regression with Distribution-Free Uncertainty Quantification and Applications in Imaging

Anastasios N. Angelopoulos*, Amit Kohli*, Stephen Bates,
Michael I. Jordan, Jitendra Malik
Department of Electrical Engineering and Computer Science, University of California, Berkeley

Thayer Alshaabi, Srigokul Upadhyayula
Advanced Bioimaging Center, Department of Molecular and Cell Biology, University of California, Berkeley

Yaniv Romano
Departments of Electrical and Computer Engineering and of Computer Science, Technion - Israel Institute of Technology

February 11, 2022

Abstract

Image-to-image regression is an important learning task, used frequently in biological imaging. Current algorithms, however, do not generally offer statistical guarantees that protect against a model's mistakes and hallucinations. To address this, we develop uncertainty quantification techniques with rigorous statistical guarantees for image-to-image regression problems. In particular, we show how to derive uncertainty intervals around each pixel that are guaranteed to contain the true value with a user-specified confidence probability. Our methods work in conjunction with any base machine learning model, such as a neural network, and endow it with formal mathematical guarantees—regardless of the true unknown data distribution or choice of model. Furthermore, they are simple to implement and computationally inexpensive. We evaluate our procedure on three image-to-image regression tasks: quantitative phase microscopy, accelerated magnetic resonance imaging, and super-resolution transmission electron microscopy of a *Drosophila melanogaster* brain.

1 Introduction

The deployment of image-to-image regression in scientific imaging has generated enormous excitement, promising a future where the resolution of an imaging system can be improved algorithmically [1]. For example, research developments in machine learning have accelerated MRI scans by an order of magnitude [2]. But, to this day, there remains an elephant in the room, obstructing the deployment of these systems: *how can we know when the model has produced an incorrect prediction?*

In most cases, we cannot. Indeed, the expressive power of modern machine learning is also its torment. Deep learning models have revolutionized predictive accuracy, but obversely, they fail in silent, unknown, and even unknowable ways. For scientific imaging settings, where learning will be used for inference and discovery, we need ways to understand when and how a model's predictions might be wrong. Nonetheless, image-to-image regression algorithms, such as those for denoising and super-resolution, are normally deployed without any notion of statistical reliability. The scientist is therefore left worrying that their new discovery is simply the model's hallucination. The purpose of this paper is to introduce a technique which rigorously quantifies the uncertainty in an image-valued point prediction, thereby alerting the scientist of potential hallucinations (see Figure 1).

We will develop a method for endowing any image-to-image regression model with per-pixel *uncertainty intervals*. At a particular pixel, an uncertainty interval is a range of values guaranteed to contain the true value of that pixel with high probability. Our contributions are the following:

*equal contribution

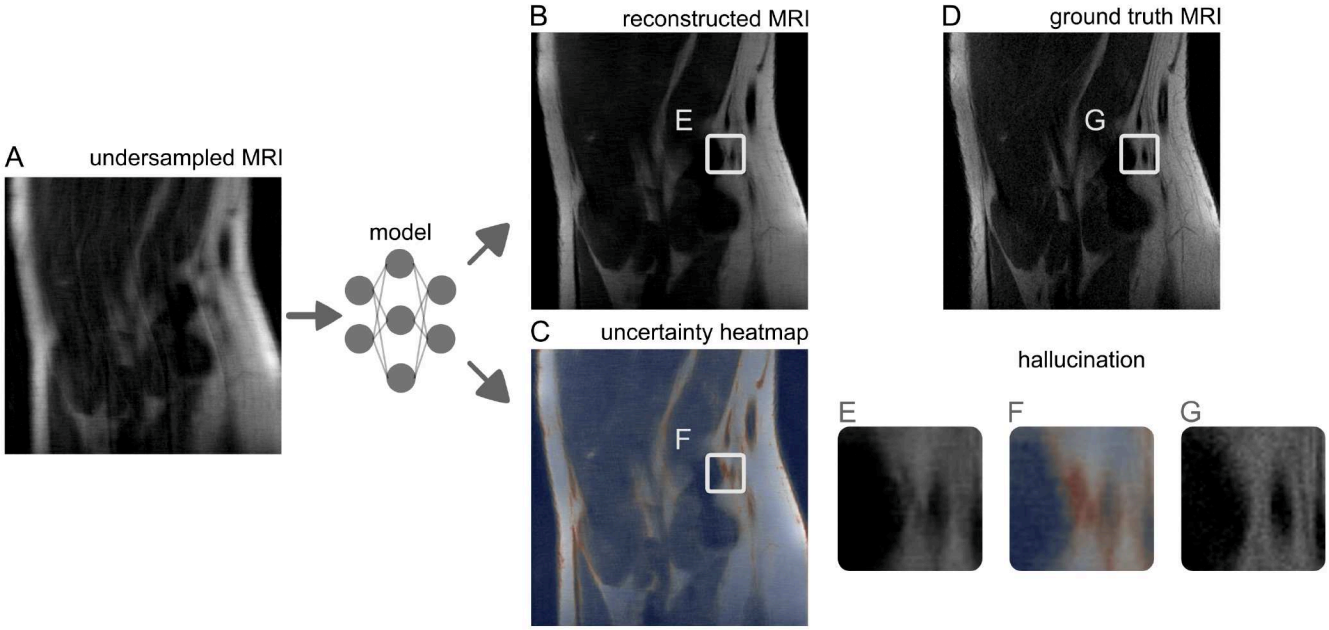


Figure 1: **An algorithmic MRI reconstruction with uncertainty.** A rapidly acquired but undersampled MR image of a knee (A) is fed into a model that predicts a sharp reconstruction (B) along with a calibrated notion of uncertainty (C). In (C), red means high uncertainty and blue means low uncertainty. Wherever the reconstruction contains hallucinations, the uncertainty is high; see the hallucination in the image patch (E), which has high uncertainty in (F), and does not exist in the ground truth (G). For experimental details, see Section 3.4.

1. We introduce *distribution-free* uncertainty quantification to image-to-image regression; this means the uncertainty intervals will have a rigorous guarantee for any image dataset and any regression model, regardless of the number of data points used to construct the interval.

2. We introduce and evaluate several practical algorithms for constructing these sets, including *pixelwise quantile regression*, an extension of quantile regression [3] to this setting. In experiments, quantile regression consistently leads to the best performance of any uncertainty quantification algorithm, often by a large amount.

3. We apply our methods to three challenging imaging problems: quantitative phase microscopy, accelerated magnetic resonance imaging (MRI), and super-resolution transmission electron microscopy of a *Drosophila melanogaster* brain. Our accompanying codebase allows easy application of these methods to any imaging problem, and the exact reproduction of the aforementioned examples.[1] The proposed calibrated pixelwise quantile regression approach offers state-of-the-art results on these tasks, in the sense that its uncertainty intervals are smaller than those from other methods.

1.1 Notation and Goal

The inputs X and outputs Y are both images in $\mathcal{X} = [0,1]^{M \times N}$ (for simplicity of notation, we discuss the case where X and Y are the same size). We also assume access to an *underlying predictor* $\hat{f}(X)$ mapping from X to a point prediction of Y. The reader can imagine X to be a downsampled version of Y, and $\hat{f}(X) \in \mathbb{R}^{M \times N}$ to be a neural network trained to upsample X and reconstruct Y (this is the *super-resolution* task).

Our task is to create uncertainty intervals around each pixel of the predicted image $\hat{f}(X)$ that contain the true pixel values with a user-specified probability. Formally, we will construct the following interval-valued function for each pixel,

$$\mathcal{T}(X)_{(m,n)} = \left[\hat{f}(X)_{(m,n)} - \hat{l}(X)_{(m,n)}, \hat{f}(X)_{(m,n)} + \hat{u}(X)_{(m,n)}\right], \tag{1}$$

[1] `https://github.com/aangelopoulos/im2im-uq`

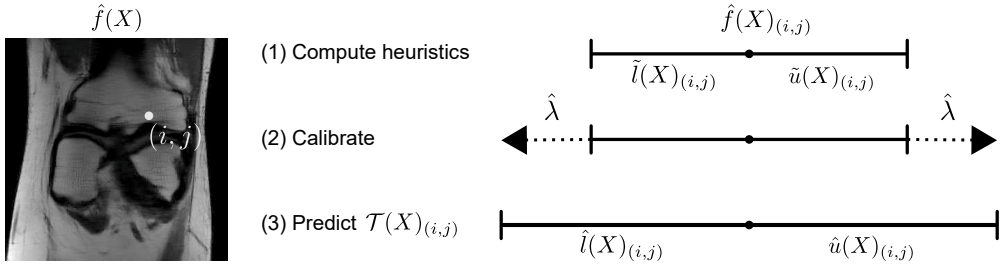


Figure 2: **An explanation of image-valued risk-controlling prediction sets.** We visualize the process of constructing an uncertainty interval for a single pixel (i, j) of the model's prediction $\hat{f}(X)$. In the first step, we compute the heuristic upper and lower interval lengths. Second, we choose $\hat{\lambda}$ via the RCPS calibration procedure in Section 2.2. Finally, we form the risk-controlling prediction set $\mathcal{T}_{\hat{\lambda}}(X)_{(i,j)}$ as in (1).

which takes an image and outputs the uncertainty interval for each pixel (m, n). Notice that the intervals always include the prediction $\hat{f}(X)$, and have width $\hat{l}(X)$ in the lower direction and $\hat{u}(x)$ in the upper direction. Intuitively, a large value in $\hat{u}(X)$ indicates a pixel that could have a much higher value than the prediction (undershooting). Likewise, a large pixel value in $\hat{l}(X)$ indicates a pixel that could have a much lower value than the prediction (overshooting). We will form the uncertainty intervals by using a held-out set of calibration data, $\{(X_i, Y_i)\}_{i=1}^n$, to assess the model's performance. The uncertainty intervals will be statistically valid in the following sense. The user selects a risk level $\alpha \in (0, 1)$, and an error level $\delta \in (0, 1)$, such as $\alpha = \delta = 0.1$. Then, we construct intervals that contain at least $1 - \alpha$ of the ground-truth pixel values with probability $1 - \delta$. That is, with probability at least $1 - \delta$,

$$\mathbb{E}\left[\frac{1}{MN}\left|\left\{(m, n) : Y^{\text{test}}_{(m,n)} \in \mathcal{T}(X^{\text{test}})_{(m,n)}\right\}\right|\right] \geq 1 - \alpha,$$

where $X^{\text{test}}, Y^{\text{test}}$ is a fresh test point from the same distribution as the calibration data.

In the next section, we will describe in detail the algorithm for generating $\hat{l}(X)$ and $\hat{u}(X)$ as well as its statistical properties. Importantly, this algorithm is modular, allowing the user to use the most complex, cutting-edge methods for learning $\hat{f}$ (i.e., the best neural network methods), all the while having uncertainty intervals that reliably communicate the quality of the predictions.

2 Methods

We now formally describe the method for constructing uncertainty intervals. Each pixel in the image will get its own uncertainty interval, as in (1), that is statistically guaranteed to contain the true value with high probability.

The procedure that yields these intervals, visualized in Figure 2, has two phases. First, we train a model to output a heuristic notion of uncertainty. In practice, this amounts to training a machine learning system to output a point prediction $\hat{f}$, a heuristic lower interval length $\tilde{l}$, and a heuristic upper interval length $\tilde{u}$ using any method, such as a neural network. In Section 2.1 we introduce and benchmark four possible methods of learning these heuristics. The uncalibrated intervals $(\hat{f}(X) - \tilde{l}(X), \hat{f}(X) + \tilde{u}(X))$ are heuristic in the sense that they do not contain the ground truth with the desired probability—we made no assumptions about the algorithm used to train $\tilde{l}$ and $\tilde{u}$. To remedy this, in the second phase we calibrate the heuristic notions of uncertainty by scaling them until they contain the right fraction of the ground truth pixels. That is, we multiply the upper and lower lengths by a value $\hat{\lambda}$ that is chosen using the procedure that we will describe in Section 2.2. The final intervals are exactly those in (1), with the upper and lower widths

$$\hat{l}(x) = \hat{\lambda}\tilde{l}(x) \quad \text{and} \quad \hat{u}(x) = \hat{\lambda}\tilde{u}(x).$$

Algorithm 1 summarizes this process.

Following the above strategy will give us uncertainty intervals that satisfy the desired statistical guarantee from Section 1.1. We call a set of these rigorous uncertainty intervals—one for each pixel in an image—an image-valued *Risk-Controlling Prediction Set*.

Algorithm 1 Generating Image-Valued RCPS

1: Train model that outputs point prediction $\hat{f}$ and heuristic lower and upper interval lengths $\tilde{l}$ and $\tilde{u}$.
2: Compute the calibrated parameter $\hat{\lambda}$ using the calibration data and Algorithm 2.
3: Construct $\mathcal{T}$ as in (1).
4: For a new image X, output the risk-controlling prediction set $\mathcal{T}(X)$.

Definition 1 (Risk-Controlling Prediction Set (RCPS), modified from [4]). *We call a random set-valued function* $\mathcal{T} : \mathcal{X} \to \left(2^{[0,1]}\right)^{M \times N}$ *an* (α,δ)-Risk-Controlling Prediction Set *if*

$$\mathbb{P}\left(\mathbb{E}\big[L(\mathcal{T}(X), Y)\big] > \alpha\right) \leq \delta, \tag{2}$$

where

$$L(\mathcal{T}(X), Y) = 1 - \frac{\left|\left\{(m,n) : Y_{(m,n)} \in \mathcal{T}(X)_{(m,n)}\right\}\right|}{MN}.$$

Remark 1. *The inner expectation in* (2) *is over a new test point,* (X, Y). *The outer probability is over the calibration data,* $\left\{(X_i, Y_i)\right\}_{i=1}^{n}$. *In other words,* $\mathcal{T}$ *is constructed based on the calibration data, which makes it a random function. We will only fail to control the risk if we are unlucky with the sample of calibration data, with probability* δ.

Parsing the above equation, we define a level α, which tells us what fraction of pixels in the image we allow to fall outside of the intervals. If we set $\alpha = 0.1$, for example, it means no more than 10% of the true pixel values will lie outside of $\mathcal{T}$ except with probability δ.

Having laid out the goal and general algorithm, we now discuss how to train the model to output heuristic notions of uncertainty for eventual calibration.

2.1 Picking a Heuristic Notion of Uncertainty

The selections of $\tilde{l}$ and $\tilde{u}$ will ultimately determine the properties of the prediction sets, such as their size and shape. We will learn these heuristics from the same training dataset used to train $\hat{f}$. Here, we develop four different heuristic notions of uncertainty, which we will evaluate and compare in later experiments (Section 3). These heuristics are

1. regression to the magnitude of the residual,
2. parameterizing each pixel as a Gaussian and reporting its standard deviation,
3. outputting a softmax distribution at each pixel, and
4. pixelwise quantile regression.

Although each of these methods is trained to predict some form of uncertainty, they may not do it well—hence the need for calibration via Algorithm 2 after training. Each heuristic requires the use of a different loss function when training the neural network via gradient descent. The remainder of this subsection describes each loss function precisely. For notational simplicity, we omit subscripts and sums indexing different pixels; in the experiments, we train our models by averaging the loss function applied to each pixel separately.

2.1.1 Magnitude of the Residual

In this flavor of uncertainty quantification, we set $\tilde{u} = \tilde{l}$, referring to both the upper and lower interval lengths as $\tilde{u}$ (the letter 'u' is a mnemonic for the 'uncertainty' of the model). We then optimize $\tilde{u}$ for the following loss function:

$$\mathcal{L}(x, y) = \left(\tilde{u}(x) - \left|\hat{f}(x) - y\right|\right)^2.$$

The loss function encourages each pixel of $\tilde{u}$ to be equal to the model's error at that pixel. Notice that $\mathcal{L}(x, y) = 0$ in the ideal case when the heuristic is exactly equal to the magnitude of the residual, i.e., $\tilde{u}(x) = |\hat{f}(x) - y|$. Estimating the magnitude of the residual is a straightforward way of quantifying a model's error, although it has two downsides. Firstly, it can only construct symmetric intervals, which makes the pixelwise intervals less informative and can inflate the set size. Second, unlike quantile regression, there is no guarantee that the residual estimate results in a valid prediction set without RCPS. Third, estimating the residual's magnitude is challenging since the training residuals are likely to be smaller than the test ones due to overfitting, unless an extra data split is used.

2.1.2 One Gaussian Per Pixel

We will now explain another common heuristic, which involves modeling each pixel as a sample from a Gaussian distribution with a particular mean and standard deviation [5]. Translating into our notation, $\hat{f}$ will be the mean function, and $\tilde{u} = \tilde{l}$ will be the standard deviation. We proceed by minimizing the negative log-likelihood of the Gaussian distribution,

$$\mathcal{L}(x, y) = \log\left(\tilde{u}(x)\right) + \frac{\left(\hat{f}(x) - y\right)^2}{\tilde{u}(x)}.$$

Like the residual magnitude method from Section 2.1.1, this heuristic is only suited to symmetric intervals and provides no guarantees of coverage without strong assumptions. Additionally, unlike the residual magnitude and quantile regression methods, one cannot use data splitting to avoid overconfidence due to overfitting.

2.1.3 Softmax Outputs

This next heuristic is most common in classification; indeed, it involves reframing image-to-image regression as a classification problem over a discrete set of pixel values. The procedure is different from the last two examples; the functions $\tilde{u}$ and $\tilde{l}$ are not equal, and they are not learned directly. Instead, we train the network to produce an entire probability distribution, and directly extract all three of $\hat{f}$, $\tilde{u}$, and $\tilde{l}$.

Let us first discretize the possible pixel values into K categories: $\{0, \frac{1}{K-1}, ..., \frac{K-1}{K-1}\}$. We then associate a discrete label with an otherwise continuous label via the function

$$D(y) = \left| \left\{ i : i \in 0, 1, ..., K-1 \text{ and } \frac{i}{K-1} \geq y \right\} \right|.$$

Intuitively, the function $D(y)$ discretizes $[0, 1]$, then bins the pixel accordingly.

This allows us to train the neural network to output distributions over pixel values $\hat{\pi}_y(x)$ estimating the conditional probabilities $\mathbb{P}[Y = y \mid X = x]$ via the cross-entropy loss,

$$\mathcal{L}(x, y) = \frac{1}{MN} \sum_{\substack{1 \leq i \leq M \\ 1 \leq j \leq N}} -\hat{\pi}_{D(y)}(x) + \log\left(\sum_{k=1}^{K} \exp\left(\hat{\pi}_{D(k)}(x)\right) \right).$$

Finally, we can extract the prediction and heuristic uncertainties,

$$\begin{aligned}
\hat{f}(x) &= \frac{1}{K-1} \arg\max_k \hat{\pi}_k(x); \\
\tilde{u}(x) &= \frac{1}{K-1} \text{Quantile}\left(1 - \frac{\alpha}{2}, \hat{\pi}(x)\right); \\
\tilde{l}(x) &= \frac{1}{K-1} \text{Quantile}\left(\frac{\alpha}{2}, \hat{\pi}(x)\right),
\end{aligned}$$

where

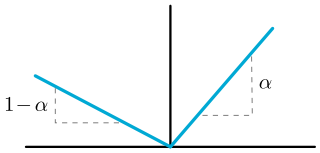
$$\text{Quantile}\left(\beta, \hat{\pi}(x)\right) = \min\left\{ K' : \sum_{k=1}^{K'} \hat{\pi}_k(x) \geq \beta \right\}.$$

The softmax approach requires discretizing $\mathcal{Y}$ into K bins, which can severely limit its utility. The heuristic can only create prediction sets whose endpoints are multiples of $1/K$, which may make it too conservative. Furthermore, the output representation can be enormous, making the memory constraints infeasible for large images (e.g., for K=256, the model produces an output of size $M \times N \times 256$).

2.1.4 Pixelwise Quantile Regression

This final heuristic is a multi-dimensional version of conformalized quantile regression [3, 6]. If we want a 90% uncertainty interval, then reporting the interval between the estimated 95% and 5% quantiles for each pixel is a valid approach. Thus, we set $\tilde{u}$ to be an estimate of the $1-\alpha/2$ conditional quantile and $\tilde{l}$ to be an estimate of the $\alpha/2$ conditional quantile. We estimate these pixelwise quantiles with a special loss function called a *quantile loss* (sometimes informally referred to as a *pinball loss*), shown below in its general form for the α quantile and its quantile estimator $\hat{q}_\alpha(x)$,

$$\mathcal{L}_\alpha\big(\hat{q}_\alpha(x), y\big) = \big(y - \hat{q}_\alpha(x)\big)\alpha \mathbb{1}\left\{y > \hat{q}_\alpha(x)\right\} + \big(\hat{q}_\alpha(x) - y\big)(1-\alpha)\mathbb{1}\left\{y \leq \hat{q}_\alpha(x)\right\}.$$



Omitting some algebra, we can see that the minimizer of this loss is the conditional quantile, i.e., $\text{Quantile}_{Y|X}(\alpha) = \min\{q : \mathbb{P}[Y < q \mid X] \leq \alpha\}$. Estimating $\hat{q}$ via empirical risk minimization should therefore approximate the conditional quantile. This can be made rigorous—under some regularity conditions, quantile regression converges asymptotically to the conditional quantile [3, 7–11]. This analysis suggests that quantile regression could be practically effective.

Note that in this case, $\tilde{u}$ and $\tilde{l}$ must be trained with different loss functions, since they estimate different quantiles. Ultimately, we collapse these into one global loss for the heuristic,

$$\mathcal{L}(x, y) = \mathcal{L}_{\alpha/2}\big(\tilde{l}(x), y\big) + \mathcal{L}_{1-\alpha/2}\big(\tilde{u}(x), y\big).$$

After training, we expect $\tilde{l}$ and $\tilde{u}$ to approximate the $\alpha/2$ and $1-\alpha/2$ quantiles respectively.

2.2 Calibrating Heuristic Notions of Uncertainty

As earlier discussed, we seek to form the RCPS in (1), which we can compute using any of the heuristics from Section 2.1. The function $\mathcal{T}$ will vary based on the heuristic notion of uncertainty used; however, the algorithm for selecting $\hat{\lambda}$ will provide the guarantee in Definition 1 regardless.

The calibration algorithm upper bounds the fraction of pixels falling outside the intervals, and then picks the smallest uncertainty intervals where the upper bound falls below α. Making this more concrete, we index the size

Algorithm 2 Pseudocode for computing $\hat{\lambda}$

Input: Calibration data, (X_i, Y_i), $i = 1, \ldots, n$; risk level α; error rate δ; underlying predictor $\hat{f}$; heuristic lower and upper interval lengths $\tilde{l}$ and $\tilde{u}$; maximum value λ_{max}; step size $d\lambda > 0$.
Output: Parameter $\hat{\lambda}$ for computing RCPS.

$\lambda \leftarrow \lambda_{\text{max}}$
$r \leftarrow 1$
while $r \leq \alpha$ **do**
  $\lambda \leftarrow \lambda - d\lambda$
  for $i = 1, ..., n$ **do**
    $L_i \leftarrow L\big(\mathcal{T}_\lambda(X_i), Y_i\big)$
  $r \leftarrow \frac{1}{n}\sum_{i=1}^{n} L_i + \sqrt{\frac{1}{2n}\log\frac{1}{\delta}}$ ▷ Can replace with any valid upper-confidence bound on the risk.
$\hat{\lambda} \leftarrow \lambda + d\lambda$ ▷ Backtrack by one because we overshot.

of the intervals with a free multiplicative factor λ,

$$\mathcal{T}_\lambda(X)_{(m,n)} = \left[\hat{f}(X)_{(m,n)} - \lambda\tilde{l}(X)_{(m,n)}, \hat{f}(X)_{(m,n)} + \lambda\tilde{u}(X)_{(m,n)}\right].$$

For a particular input image, when λ grows, the intervals grow; for a sufficiently large λ, the intervals will contain all of the ground truth pixel values. Our job is to pick $\hat{\lambda}$ to be the smallest value such that $\mathcal{T}_{\hat{\lambda}}(X)$ satisfies Definition 1 (note that $\mathcal{T}(X) = \mathcal{T}_{\hat{\lambda}}(X)$). Using the calibration dataset, we form Hoeffding's upper-confidence bound,

$$\widehat{R}^+(\lambda) = \frac{1}{n}\sum_{i=1}^{n} L(\mathcal{T}_\lambda(X_i), Y_i) + \sqrt{\frac{1}{2n}\log\frac{1}{\delta}}.$$

It is shown in [12] that the Hoeffding bound is valid, that is, $\mathbb{P}\left[\widehat{R}^+(\lambda) < R(\lambda)\right] < \delta$. Knowing this, we can use $\widehat{R}^+(\lambda)$ to pick the smallest λ satisfying Definition 1. There is a closed-form expression for this process,

$$\hat{\lambda} = \min\left\{\lambda : \widehat{R}^+(\lambda') \leq \alpha, \ \ \forall \alpha' \geq \alpha\right\}. \tag{3}$$

Proposition 1 ($\mathcal{T}_{\hat{\lambda}}$ is an RCPS [4])**.** *With $\hat{\lambda}$ selected as in* (3)*, $\mathcal{T}_{\hat{\lambda}}$ satisfies Definition 1.*

For the proof of this fact, along with a discussion of the tighter confidence bounds used in the experiments, see previous work [4, 13]. This calibration procedure is easy to implement in code, and we summarize it in Algorithm 2.

3 Experiments

The following sequence of experiments applies our methods to several challenging settings in biological imaging. The goal of these experiments is twofold. First, we demonstrate the utility of the procedures in practical experiments. Second, we evaluate the comparative effectiveness of the different heuristics qualitatively, as well as with a series of quantitative metrics. We will briefly discuss these metrics before providing the details of each experiment.

3.1 Evaluation Metrics

Empirical risk. The first quantity to notice is the risk, which should fall below α with probability $(1 - \delta)$. This is guaranteed in general by Proposition 1. For each dataset and heuristic, we make a histogram of the risk over several runs of the RCPS calibration, showing it is indeed controlled at the desired level.

Prediction set size. If the underlying heuristic notion of uncertainty is poor, then, in order to control the risk, the sets may need to be large. Generally, such an output is not informative to a practitioner, and all else equal, smaller intervals give more actionable assessments of the regression's quality. Thus, we report histograms of the interval size for each metric—smaller is better.

Size-stratified risk. Next, we seek prediction sets that do not systematically make mistakes in difficult parts of the image. Our risk control requirement in Definition 1 may be satisfied even if the prediction sets systematically fail to contain the most difficult pixels. For example, if $\alpha = 0.1$ and 90% of pixels are covered by fixed-width intervals of size 0.01, then the requirement is satisfied—however, the sets no longer serve as useful notions of uncertainty. To investigate such behavior, we evaluate the *size-stratified risk* [14]—i.e., we stratify pixels by the quartile of their interval sizes, and report the empirical risk within each of these quartiles. The desire is to have the risk be at approximately the same level for all strata, i.e., the risk should be as similar as possible between pixels with different set sizes. In other words, when we make a barplot of the stratified risk, the bars should all be the same height. Achieving this balance means the algorithm is not over-including easier-to-estimate pixels in order to excuse poor performance on difficult ones.

MSE of point prediction. Finally, we want to pick a heuristic notion of uncertainty that does not harm the accuracy of the point prediction during the joint training process. To measure this, we plot the *mean-squared error* (MSE) on the validation set for the point prediction which was jointly trained with each heuristic measure of uncertainty. A lower mean-squared error means that the joint training of the point prediction and heuristic uncertainty worked nicely, and did not degrade the point prediction. For certain heuristics, such as the Gaussian

and softmax versions, this measure is particularly important because these methods do not directly optimize for the MSE and instead require a different procedure for supervising the point prediction (maximizing the Gaussian log likelihood and minimizing the cross-entropy loss, respectively).

Visualizations. In addition to the quantitative metrics, there is no substitute for seeing visualizations of the uncertainty intervals. For each example, we show the input, output, and the prediction sets generated by quantile regression along with the ground truth target. We represent the prediction sets by passing the pixelwise interval lengths through a colormap, where small sets render a pixel blue and large sets render it red. The interpretation, then, is that the redder a region is, the more uncertain it is and conversely, the bluer a region is, the more confident it is. Consequently, we expect the colormap to be red where the model is missing biological features and around fine structures such as edges which are difficult to reconstruct from partial data.

3.2 Experimental Details

We use a standard experimental pipeline for all of the forthcoming experiments. In all experiments, we fit the predictor $\hat{f}$ and the heuristic notions of uncertainty $\tilde{u}$ and $\tilde{l}$ jointly. To ensure a level playing field and to promote reproducibility, the code used to define, train, and evaluate the model is shared among all heuristics and datasets. In order to run a new experiment (e.g., on a new dataset or with a new heuristic), minimal additional code is needed. We believe this code to be a primary contribution of this paper with utility to researchers and practitioners alike. It lives at the following open-sourced GitHub link: https://github.com/aangelopoulos/im2im-uq.

In each experiment, an 8-layer U-Net [15] is used as the base model architecture and trained with an Adam optimizer for 10 epochs. We swept over two learning rates, $\{0.001, 0.0001\}$, and chose the learning rate that minimized the point prediction's MSE for each method in each experiment. All images get normalized to the interval $[0, 1]$. For the softmax heuristic, we discretized the prediction space with $K = 50$ because larger choices of K become too computationally expensive due to the amount of memory needed to store the extra dimension—a major practical limitation of this heuristic. We choose $\alpha = \delta = 0.1$ for the RCPS procedure in all cases, and adaptively select a grid of 1000 values of λ for each experiment. We evaluate each method by plotting its risk, average set size, size-stratified risk, and mean-squared error of the jointly trained prediction, as well as displaying an example. Further experimental details are available in the codebase.

We now discuss each experiment in detail. For each experiment, we include a brief background of the imaging problem, a description of the inputs X and outputs Y, and the aforementioned evaluation metrics for each heuristic.

3.3 Quantitative Phase Microscopy of Leukocytes

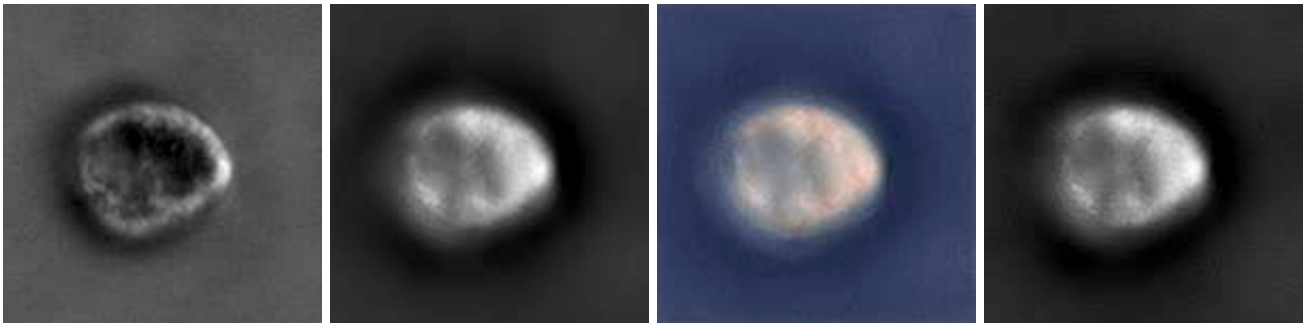


Figure 3: **Examples of quantitative phase reconstructions with uncertainty** shown in the following order: input, prediction, uncertainty visualization, ground truth. We use the quantile regression version of the procedure.

Background. In order to image the structure of cells—which are essentially transparent bags of water—one must measure their local refractive index, or equivalently, the phase delay incurred by light passing through each region of the cell. This task, known as quantitative phase imaging or QPI [16–18], requires an algorithm to map intensity-only images to the phase value at each pixel, since it is impossible to directly measure the phase of light. Generally, as input to the algorithm, these methods take in a diverse set of intensity images captured under different imaging conditions, such as the illumination angle. Their performance improves with more input images.

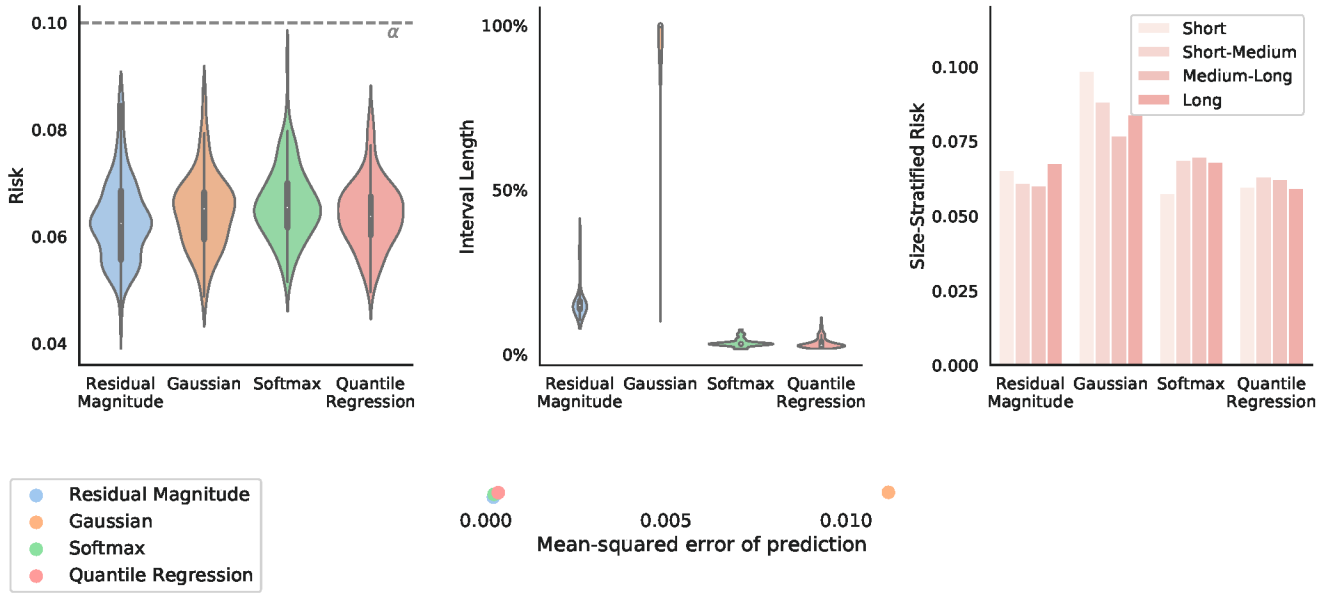


Figure 4: **A quantitative summary of all four heuristics** after RCPS calibration in the quantitative phase example. All methods control the risk, and quantile regression has the smallest set size. The Gaussian method has poor MSE, interval size, and size-stratified risk because it did not converge in training for either of the learning rates we chose.

Dataset description. Framing QPI as image-to-image regression, we take the input X to be the concatenation of two obliquely illuminated cell intensity images (from opposite angles) and the target Y to be a reference phase image. Y is obtained using an analytic phase recovery technique known as differential phase contrast (DPC), which takes four or more images as input [19]. The utility, then, of the regression model is to reduce the number of input images needed for high quality phase prediction, thereby improving the tradeoff between acquisition speed and prediction accuracy. Furthermore, the quantitative phase values have an intrinsic meaning, so by adding uncertainty intervals which are on the scale of the phase values, we provide an important inferential tool for analyzing cell morphology.

For the experiment, we use the Berkeley Single Cell Computational Microscopy (BSCCM) dataset [20], which contains 2,000 single leukocyte (white blood cell) images with 150x150 pixels taken using several imaging modalities. Of particular interest to us, this dataset includes images taken under a variety of different angles of illumination co-registered with quantitative phase maps obtained via 4-image DPC. As input to the U-Net, we concatenate two obliquely illuminated cell images along the channel dimension. We use 1800 randomly selected data points with a batch size of 64 to train the model, 100 points for calibration, and 100 points for validation. Our results are visualized in Figure 3.

Results. We report our results in Figure 5. As promised by the calibration procedure, risk-control holds for all choices of heuristic uncertainties. In terms of statistical power, we see that quantile regression outcompetes the other heuristics in the trifecta of evaluations—it has the smallest average set size and best size-stratified coverage while remaining competitive with other methods in mean-squared error. Altogether, these metrics express that the uncertainty intervals produced by calibrated quantile regression are tight and adaptive to the model's performance, even among different pixels within a single prediction. The softmax heuristic, though seemingly competitive in these evaluations, gives nearly fixed-width intervals, most of which have exactly the same size because of the discretization.

3.4 Fast Magnetic Resonance Imaging

Background. Much like our previous example, in MRI there exists a tradeoff between imaging speed and quality. MRI directly samples an object's spatial frequency (k-space) over time; so it is possible to reduce the scan time by lowering the effective sampling rate in k-space. Although fast imaging is more comfortable for human subjects and also critical for certain fast movements like the beating of the heart, insufficient sampling results in low quality,

aliased MR images. However, with deep learning, we can try to fill in the information lost by undersampling to emulate fully sampled images, thereby getting the joint benefits of fast scan times and high quality reconstructions.

Dataset description. The inputs X are the undersampled images formed by downsampling k-space by a factor of four along a single dimension (the phase encoding direction), and then taking an inverse Fourier transform. Our outputs Y are the fully sampled MR images. Successfully regressing X to Y essentially accelerates the MRI scan time by a factor of four.

We use the FastMRI dataset for this example [2]. The dataset includes 10,000 clinical knee MR volumes taken with 3T or 1.5T magnets which are algorithmically undersampled with k-space masks that emulate fast sampling strategies. We dissect the volumes into 27,993 randomly selected 320x320 pixel coronal knee slices for training the model, 3,474 for the RCPS calibration, and 3,474 for validation. We use a batch size of 78. For the Gaussian method, we standardized the output space to be mean zero and unit variance, since it failed to properly train when normalized to fall in the interval $[0, 1]$.

Results. Qualitatively, Figure 1 shows an example of an MRI reconstruction using calibrated quantile regression. The predictions are slightly blurred versions of the ground truth, likely due to the network's bias toward low frequency outputs [21]. The uncertainty intervals have large values in areas with high contrast, expressing the intrinsic uncertainty in localizing edges using incomplete information. The quantitative results of this experiment, visualized in Figure 5, are in line with those of the QPI experiment; we achieve the desired risk level and quantile regression performs best on all metrics. Although the softmax heuristic has near-even size-stratified risk, this is because it outputs quantized sets of nearly fixed size, and the strata are therefore decided by random tie-breaking.

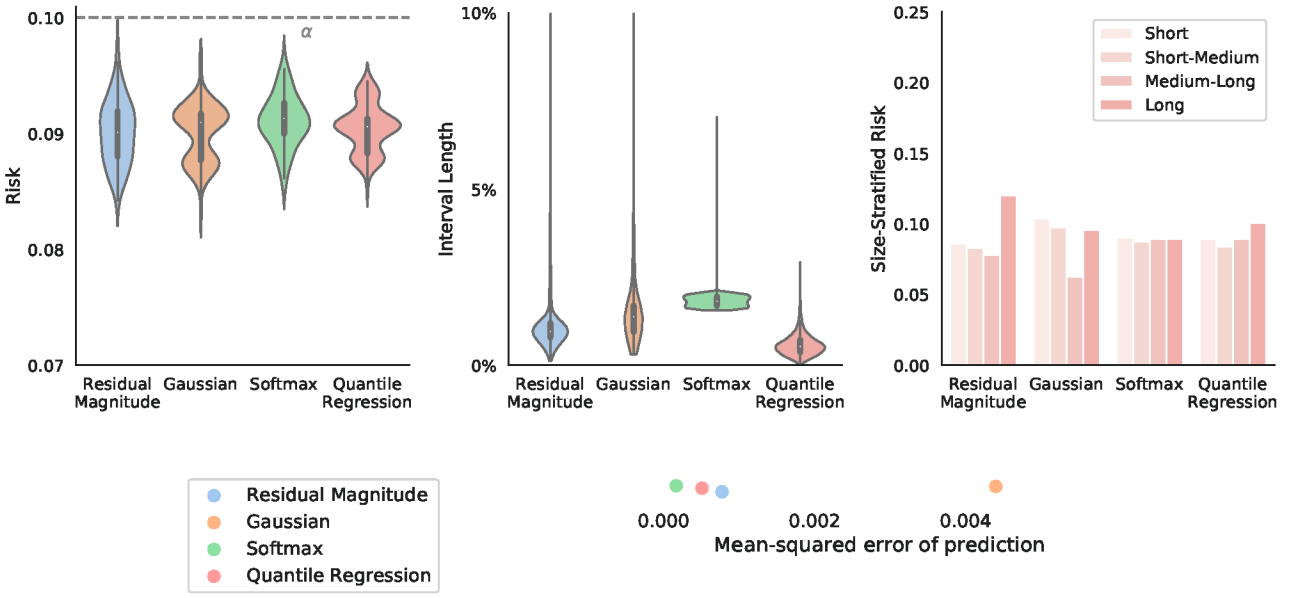


Figure 5: **A quantitative summary of all four heuristics** after RCPS calibration in the FastMRI example. All three methods control the risk. Quantile regression has the smallest interval size and best size-stratified risk.

3.5 Drosophila Brain Transmission Electron Microscopy

Background. Finally, we perform algorithmic super-resolution transmission electron microscopy (TEM) of the brain of a Drosophila melanogaster (fruit fly). TEM uses focused electron beams rather than visible light to produce images, and due to the small de Broglie wavelength of electrons, it can achieve significantly higher resolution than visible light microscopy, on the order of single nanometers. However, TEM sequentially scans over the sample volume, imaging point-by-point; thus, its scan time scales cubically with the desired resolution. For large volumes like the fly brain shown in Figure 6, imaging can take years. Upsampling lower resolution (say 16nm) TEM data to high resolution (4nm) images could therefore save months of time.

Dataset description. We consider super-resolution as image-to-image regression, where X is a manually down-sampled version of a 4nm TEM image Y. In particular, we consider a factor of 4x nearest-neighbor downsampling

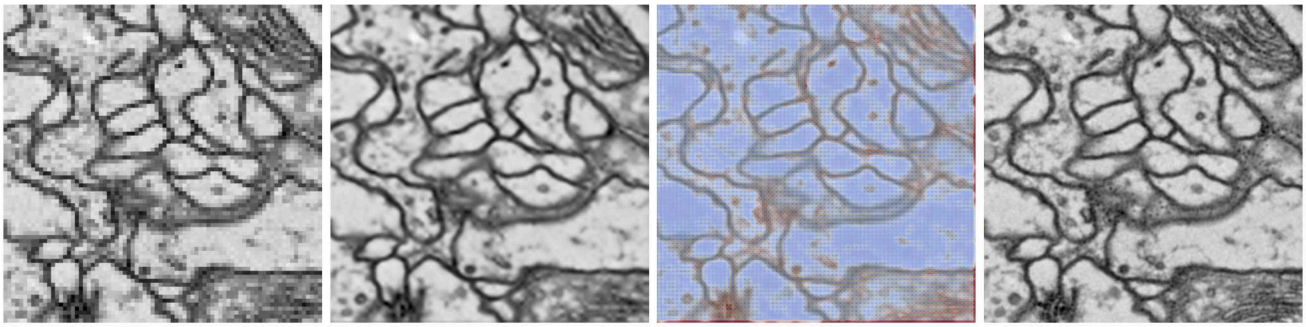


Figure 6: **Examples of Drosophila brain reconstructions with uncertainty** shown in the following order: input, prediction, uncertainty visualization, ground truth. We use the pixelwise quantile regression version of the procedure.

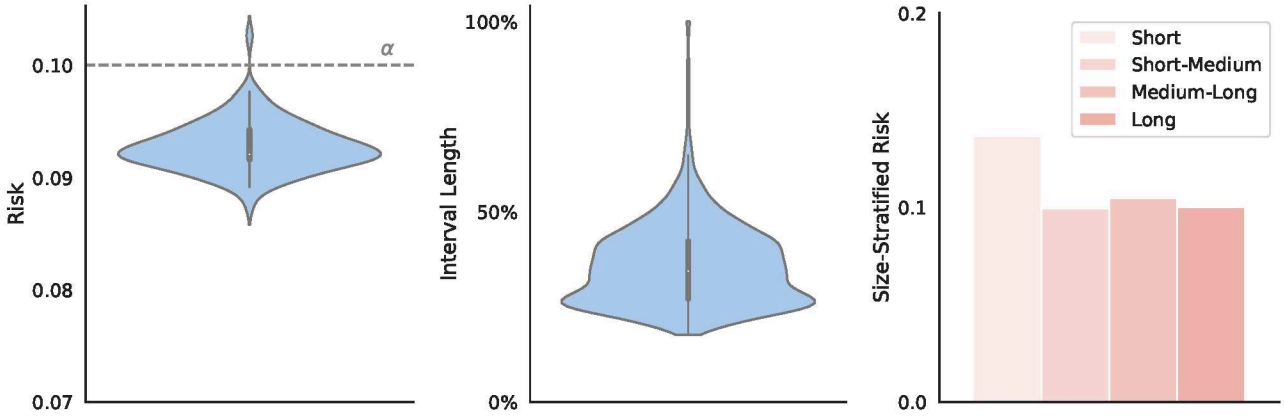


Figure 7: **Quantitative results of pixelwise quantile regression** on the TEMCA fly brain dataset. The risk is controlled, the intervals have reasonable lengths, and the size-stratified risk is roughly balanced, although slightly more permissive with small intervals.

along both image dimensions to emulate the acquisition of a 16nm TEM image.

We use the Janelia Transmission Electron Microscopy Camera Array (TEMCA2) dataset of the Full Adult Fly Brain [22]. This dataset contains a 26 TB fly brain volume at four nanometer resolution isotropically along all dimensions. As a consequence of this data burden, we did not run the full suite of procedures on the dataset, and instead ran only the consistently best performing method—pixelwise quantile regression. The dataset cannot be stored normally and must be chunked *on-the-fly* into patches, so exact dataset sizes cannot be known in advance. We used roughly 2M images of size 320x320 for training, 25K images for calibration, and 25K images for validation. We use a batch size of 16. Each image is only seen once.

Results. The results of this procedure are depicted in Figure 6. The quantitative measures in Figure 7 are similar to those of past experiments. Qualitatively, the prediction sets identify regions of high contrast as more uncertain, perhaps due to the spectral bias of CNNs. The sets are periodically zero-length, i.e., fully confident, every four pixels. This is because the input is a downsampled version of the target, so the model perfectly knows every fourth pixel. Consequently, at those pixels, the model does not have any uncertainty. This highlights the adaptivity and tightness of the prediction sets; they are not only useful in understanding where the model is poor, but also where the model performs reliably.

4 Discussion

The experimental findings indicate that pixelwise quantile regression has strong performance while avoiding many of the drawbacks of other methods. The softmax method is prohibitively memory-intensive, while achieving slightly

worse results than quantile regression in Section 3.3. The Gaussian method, all else equal, seems more difficult to train, and the `GaussianNLL` loss adversely affects both the prediction quality and the heuristic. The two winners seem to be regression to the magnitude of the residual and pixelwise quantile regression. Ultimately, we suggest pixelwise quantile regression because it supports asymmetric intervals, while also achieving better empirical performance. We hope our sequence of large-scale imaging examples indicate the broad utility of the techniques.

5 Related Work

Image-to-image regression. The problem of image-to-image regression has existed for several decades under a variety of names. The most fundamental class of image-to-image regression problems involves interpolating between samples of a digital image, a setting currently referred to as *image super-resolution*. Since this problem simplifies to interpolating between discrete samples of a two-dimensional function, methods such as linear interpolation, used as early as 200BC in *The Nine Chapters on the Mathematical Art* [23, 24] and 200AD in Ptolemy's *Almagest* [25], remain commonly deployed today. Similarly, over a hundred years of research on optimal interpolation, such as that on spline approximations [26–28], has been used in image super-resolution since the 1980s [29] and continues to be applied. In the 21st century, learning-based approaches [30–32] have dominated the research conversation, particularly those using convolutional neural networks [33–35] and generative adversarial networks [36, 37]. Beyond interpolation, image-to-image regression also encompasses denoising [38–42], style transfer [43–46], image colorization [47], and so on. A line of work based on the U-Net [15] adapts the above techniques for biomedical imaging problems, achieving strong results [48, 49]. We build directly on this line of work.

Heuristic notions of uncertainty. The idea of assuming the output of a neural network has a Gaussian distribution and maximizing its log-likelihood with gradient descent has been employed since at least 1994 [5]. Although the idea of estimating the magnitude of the residual with a Gaussian process been suggested [50], we are unaware of any papers that use the exact formulation in Section 2.1.1, and this is far from a standard method. The idea of the cross-entropy loss leading to the softmax distributional estimate has its roots in the Kraft-McMillan theorem [51, 52] and related information-theoretic concepts [53]. Quantile regression was proposed in the mid-1970s by Koenker and Bassett [3]. Since then, many papers have used the technique, applying it to economics [54–56], machine learning [57–59], medical research [60], and more. A large and vibrant community continues to work today on quantile regression, exploring variations of the technique and their operating characteristics under various conditions, such as for local polynomials [7], in additive models [61], for improved conditional coverage [62], and in deep learning [63]. Unlike the other heuristics discussed, quantile regression comes with an asymptotic guarantee of conditional coverage under certain weak regularity conditions [3, 7–11]. Accessible and complete references to the topic of quantile regression are provided in the references [64, 65]. The calibrated, pixelwise version of quantile regression extends these procedures to the image-to-image regression case. A large literature on Bayesian and approximately Bayesian uncertainty quantification also exists, including MC-Dropout [66]. Ensemble methods [63, 67, 68] are also common for uncertainty quantification in deep learning. These heuristics, which do not have finite-sample guarantees, fall outside the scope of our discussion, and we refer the reader to [69] for an introduction to that area.

Distribution-Free Uncertainty Quantification. Conformal prediction is a general, lightweight procedure for creating uncertainty intervals from any heuristic with finite-sample coverage while requiring no model retraining [70–75]. Of particular interest to us is the method of conformalized quantile regression (CQR) [6]. We directly build on CQR, by replacing the the conformal subroutine with the fixed-sequence testing procedure from [4, 13]. Other works have applied distribution-free uncertainty quantification to biological and medical computer vision tasks [76–81]. However, we are not aware of any that have studied image-to-image regression. An introduction to these topics is available in [14].

References

[1] M. Weigert, U. Schmidt, T. Boothe, A. Müller, A. Dibrov, A. Jain, B. Wilhelm, D. Schmidt, C. Broaddus, S. Culley, *et al.*, "Content-aware image restoration: Pushing the limits of fluorescence microscopy," *Nature Methods*, vol. 15, no. 12, pp. 1090–1097, 2018.

[2] J. Zbontar, F. Knoll, A. Sriram, T. Murrell, Z. Huang, M. J. Muckley, A. Defazio, R. Stern, P. Johnson, M. Bruno, *et al.*, "fastMRI: An open dataset and benchmarks for accelerated MRI," *arXiv preprint arXiv:1811.08839*, 2018.

[3] R. Koenker and G. Bassett Jr, "Regression quantiles," *Econometrica: Journal of the Econometric Society*, vol. 46, no. 1, pp. 33–50, 1978.

[4] S. Bates, A. Angelopoulos, L. Lei, J. Malik, and M. Jordan, "Distribution-free, risk-controlling prediction sets," *Journal of the Association for Computing Machinery*, vol. 68, no. 6, Sep. 2021.

[5] D. Nix and A. Weigend, "Estimating the mean and variance of the target probability distribution," in *IEEE International Conference on Neural Networks*, vol. 1, 1994, pp. 55–60.

[6] Y. Romano, E. Patterson, and E. Candès, "Conformalized quantile regression," in *Advances in Neural Information Processing Systems*, vol. 32, 2019, pp. 3543–3553.

[7] P. Chaudhuri, "Global nonparametric estimation of conditional quantile functions and their derivatives," *Journal of Multivariate Analysis*, vol. 39, no. 2, pp. 246–269, 1991.

[8] I. Steinwart and A. Christmann, "Estimating conditional quantiles with the help of the pinball loss," *Bernoulli*, vol. 17, no. 1, pp. 211–225, 2011.

[9] I. Takeuchi, Q. V. Le, T. D. Sears, and A. J. Smola, "Nonparametric quantile estimation," *Journal of Machine Learning Research*, vol. 7, pp. 1231–1264, 2006.

[10] K. Q. Zhou, S. L. Portnoy, *et al.*, "Direct use of regression quantiles to construct confidence sets in linear models," *The Annals of Statistics*, vol. 24, no. 1, pp. 287–306, 1996.

[11] K. Q. Zhou and S. L. Portnoy, "Statistical inference on heteroscedastic models based on regression quantiles," *Journal of Nonparametric Statistics*, vol. 9, no. 3, pp. 239–260, 1998.

[12] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *Journal of the American Statistical Association*, vol. 58, no. 301, pp. 13–30, 1963.

[13] A. N. Angelopoulos, S. Bates, E. J. Candès, M. I. Jordan, and L. Lei, "Learn then test: Calibrating predictive algorithms to achieve risk control," *arXiv preprint arXiv:2110.01052*, 2021.

[14] A. N. Angelopoulos and S. Bates, "A gentle introduction to conformal prediction and distribution-free uncertainty quantification," *arXiv preprint arXiv:2107.07511*, 2021.

[15] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2015, pp. 234–241.

[16] M. Mir, B. Bhaduri, R. Wang, R. Zhu, and G. Popescu, "Quantitative phase imaging," *Progress in Optics*, vol. 57, no. 133-37, p. 217, 2012.

[17] Y. Park, C. Depeursinge, and G. Popescu, "Quantitative phase imaging in biomedicine," *Nature Photonics*, vol. 12, no. 10, pp. 578–589, 2018.

[18] Y. Jo, H. Cho, S. Y. Lee, G. Choi, G. Kim, H. Min, and Y. Park, "Quantitative phase imaging and artificial intelligence: A review," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 25, no. 1, pp. 1–14, 2018.

[19] L. Tian and L. Waller, "Quantitative differential phase contrast imaging in an led array microscope," *Optics Express*, vol. 23, no. 9, pp. 11 394–11 403, 2015.

[20] H. Pinkard, *Berkeley single cell computational microscopy dataset*, 2021.

[21] N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. Hamprecht, Y. Bengio, and A. Courville, "On the spectral bias of neural networks," in *International Conference on Machine Learning*, vol. 97, PMLR, 2019, pp. 5301–5310.

[22] Z. Zheng, J. S. Lauritzen, E. Perlman, C. G. Robinson, M. Nichols, D. Milkie, O. Torrens, J. Price, C. B. Fisher, N. Sharifi, *et al.*, "A complete electron microscopy volume of the brain of adult Drosophila melanogaster," *Cell*, vol. 174, no. 3, pp. 730–743, 2018.

[23] Generations of Chinese mathematicians, *The Nine Chapters on the Mathematical Art*. 200BC-100AD.

[24] J. Needham and L. Gwei-Djen, *Science and Civilisation in China: Volume 3, Mathematics and the Sciences of the Heavens and the Earth*. Cambridge University Press, 1959.

[25] Ptolemy, *Amagest*. 200AD.

[26] H. Hahn, "Über das interpolationsproblems," *Mathematische Zeitschrift*, vol. 1, pp. 115–142, 1918.

[27] I. J. Schoenberg and A. Whitney, "On Pólya frequency functions III. the positivity of translation determinants whith an application to the interpolation problem by spline curves," *Transactions of the American Mathematical Society*, vol. 74, pp. 246–259, 1953.

[28] J. Walsh, J. Ahlberg, and E. Nilson, "Best approximation properties of the spline fit," *Journal of Mathematics and Mechanics*, vol. 11, no. 2, pp. 225–234, 1962.

[29] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, no. 6, pp. 1153–1160, 1981.

[30] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Computer Graphics and Applications*, vol. 22, no. 2, pp. 56–65, 2002.

[31] H. Chang, D.-Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, vol. 1, 2004, pp. I–I.

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

[33] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European Conference on Computer Vision*, Springer, 2014, pp. 184–199.

[34] ——, "Image super-resolution using deep convolutional networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, 2015.

[35] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European Conference on Computer Vision*, Springer, 2016, pp. 694–711.

[36] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[37] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4681–4690.

[38] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," *Multiscale Modeling and Simulation*, vol. 4, no. 2, pp. 490–530, 2005.

[39] ——, "A non-local algorithm for image denoising," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2005, pp. 60–65.

[40] H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with BM3D?" In *IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2012, pp. 2392–2399.

[41] C. Tian, L. Fei, W. Zheng, Y. Xu, W. Zuo, and C. Lin, "Deep learning on image denoising: An overview," *Neural Networks*, vol. 131, pp. 251–275, 2020.

[42] B. Goyal, A. Dogra, S. Agrawal, B. S. Sohi, and A. Sharma, "Image denoising review: From classical to state-of-the-art approaches," *Information Fusion*, vol. 55, pp. 220–244, 2020.

[43] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2414–2423.

[44] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1125–1134.

[45] J. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *IEEE International Conference on Computer Vision*, 2017, pp. 2223–2232.

[46] Y. Jing, Y. Yang, Z. Feng, J. Ye, Y. Yu, and M. Song, "Neural style transfer: A review," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 11, pp. 3365–3385, 2020.

[47] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *European Conference on Computer Vision*, Springer, 2016, pp. 649–666.

[48] J. Zbontar, F. Knoll, A. Sriram, T. Murrell, Z. Huang, M. J. Muckley, A. Defazio, R. Stern, P. Johnson, M. Bruno, *et al.*, "Attention U-Net: Learning where to look for the pancreas," *Conference on Medical Imaging with Deep Learning*, 2018.

[49] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, "U-Net++: A nested U-Net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, D. Stoyanov, Z. Taylor, G. Carneiro, T. Syeda-Mahmood, A. Martel, L. Maier-Hein, J. M. R. Tavares, A. Bradley, J. P. Papa, V. Belagiannis, J. C. Nascimento, Z. Lu, S. Conjeti, M. Moradi, H. Greenspan, and A. Madabhushi, Eds., Cham: Springer International Publishing, 2018, pp. 3–11.

[50] X. Qiu, E. Meyerson, and R. Miikkulainen, "Quantifying point-prediction uncertainty in neural networks via residual estimation with an i/o kernel," in *International Conference on Learning Representations*, 2020.

[51] L. G. Kraft, "A device for quantizing, grouping, and coding amplitude-modulated pulses," Ph.D. dissertation, Massachusetts Institute of Technology, 1949.

[52] B. McMillan, "Two inequalities implied by unique decipherability," *IRE Transactions on Information Theory*, vol. 2, no. 4, pp. 115–116, 1956.

[53] J. A. Thomas and T. M. Cover, *Elements of information theory*. John Wiley & Sons, 1999.

[54] P. K. Chaney, M. Faccio, and D. Parsley, "The quality of accounting information in politically connected firms," *Journal of Accounting and Economics*, vol. 51, no. 1-2, pp. 58–76, 2011.

[55] D. McKenzie and H. Rapoport, "Network effects and the dynamics of migration and inequality: Theory and evidence from Mexico," *Journal of Development Economics*, vol. 84, no. 1, pp. 1–24, 2007.

[56] J. A. Machado and J. Mata, "Counterfactual decomposition of changes in wage distributions using quantile regression," *Journal of Applied Econometrics*, vol. 20, no. 4, pp. 445–465, 2005.

[57] C. Hwang and J. Shim, "A simple quantile regression via support vector machine," in *International Conference on Natural Computation*, Springer, 2005, pp. 512–520.

[58] N. Meinshausen and G. Ridgeway, "Quantile regression forests," *Journal of Machine Learning Research*, vol. 7, no. 6, 2006.

[59] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers in Neurorobotics*, vol. 7, p. 21, 2013.

[60] P. Armitage, G. Berry, and J. N. S. Matthews, *Statistical methods in medical research*. John Wiley & Sons, 2008.

[61] R. Koenker, "Additive models for quantile regression: Model selection and confidence bandaids," *Brazilian Journal of Probability and Statistics*, vol. 25, no. 3, pp. 239–262, 2011.

[62] S. Feldman, S. Bates, and Y. Romano, "Improving conditional coverage via orthogonal quantile regression," in *Advances in Neural Information Processing Systems*, 2021.

[63] S. Fort, H. Hu, and B. Lakshminarayanan, "Deep ensembles: A loss landscape perspective," *arXiv preprint arXiv:1912.02757*, 2019.

[64] R. Koenker, *Quantile Regression*. Cambridge University Press, 2005.

[65] R. Koenker, V. Chernozhukov, X. He, and L. Peng, "Handbook of quantile regression," 2018.

[66] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *International Conference on Machine Learning*, PMLR, 2016, pp. 1050–1059.

[67] L. K. Hansen and P. Salamon, "Neural network ensembles," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 10, pp. 993–1001, 1990.

[68] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *Advances in Neural Information Processing Systems*, 2017, pp. 6402–6413.

[69] Y. Gal, "Uncertainty in deep learning," Ph.D. dissertation, University of Cambridge, 2016.

[70] V. Vovk, A. Gammerman, and C. Saunders, "Machine-learning applications of algorithmic randomness," in *International Conference on Machine Learning*, 1999, pp. 444–453.

[71] V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic Learning in a Random World*. New York, NY, USA: Springer, 2005.

[72] J. Lei, A. Rinaldo, and L. Wasserman, "A conformal prediction approach to explore functional data," *Annals of Mathematics and Artificial Intelligence*, vol. 74, pp. 29–43, 2015.

[73] J. Lei, J. Robins, and L. Wasserman, "Distribution-free prediction sets," *Journal of the American Statistical Association*, vol. 108, no. 501, pp. 278–287, 2013.

[74] J. Lei, "Classification with confidence," *Biometrika*, vol. 101, no. 4, pp. 755–769, Oct. 2014.

[75] M. Sadinle, J. Lei, and L. Wasserman, "Least ambiguous set-valued classifiers with bounded error levels," *Journal of the American Statistical Association*, vol. 114, pp. 223–234, 2019.

[76] Y. Hechtlinger, B. Póczos, and L. Wasserman, "Cautious deep learning," *arXiv preprint arXiv:1805.09460*, 2018.

[77] M. Cauchois, S. Gupta, and J. C. Duchi, "Knowing what you know: Valid and validated confidence sets in multiclass and multilabel prediction," *Journal of Machine Learning Research*, vol. 22, no. 81, pp. 1–42, 2021.

[78] Y. Romano, M. Sesia, and E. Candès, "Classification with valid and adaptive coverage," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., 2020, pp. 3581–3591.

[79] A. N. Angelopoulos, S. Bates, J. Malik, and M. I. Jordan, "Uncertainty sets for image classifiers using conformal prediction," in *International Conference on Learning Representations (ICLR)*, 2021.

[80] A. N. Angelopoulos, S. Bates, T. Zrnic, and M. I. Jordan, "Private prediction sets," *arXiv preprint arXiv:2102.06202*, 2021.

[81] C. Lu, A. Lemay, K. Chang, K. Hoebel, and J. Kalpathy-Cramer, "Fair conformal predictors for applications in medical imaging," *arXiv preprint arXiv:2109.04392*, 2021.