SaccadeCam: Adaptive Visual Attention for Monocular Depth Sensing

Brevin Tilmon   Sanjeev J. Koppal

University of Florida

Abstract

Most monocular depth sensing methods use convention-ally captured images that are created without considering scene content. In contrast, animal eyes have fast mechanical motions, called saccades*, that control how the scene is imaged by the fovea, where resolution is highest. In this paper, we present the SaccadeCam framework for adaptively distributing resolution onto regions of interest in the scene. Our algorithm for adaptive resolution is a self-supervised network and we demonstrate results for end-to-end learning for monocular depth estimation. We also show preliminary results with a real SaccadeCam hardware prototype.*

1. Introduction

Deep depth estimation from a single view has been effective at demonstrating the rich geometric cues available in an image [51, 49, 36, 52, 10]. Additionally, these results are improved by using other cues, such as sparse LIDAR or stereo measurements [55, 66, 35, 6]. Our key idea is to notice that most previous monocular approaches assume a nearly equal distribution of sensor pixels across the camera's field-of-view (FOV). In contrast, animal eyes distribute resolution unevenly using fast, mechanical motion, or *saccades*, that change where the eye's fovea views the scene with high acuity. In this paper, we present *Saccade-Cam*, a new algorithmic and hardware framework for visual attention control that automatically distributes resolution onto a scene to improve monocular depth estimation.

1.1. Why Leverage Attention for Depth Sensing?

Many methods seek to replicate the biological advantages of attention, such as computational efficiency. However, most efforts apply attention within network training and testing, *after* images have been captured [48, 59, 34, 63, 29]. Our framework complements existing attention-based learning, since SaccadeCam leverages visual attention to distribute resolution *during* image capture, and deep attention mechanisms can still be applied after the capture of a SaccadeCam image. Since SaccadeCam can leverage
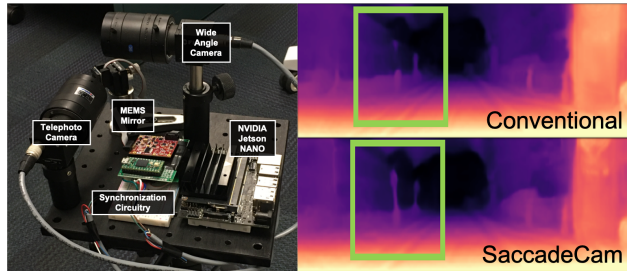

Figure 1: Our method learns to distribute resolution onto regions that improve self-supervised monocular depth estimation while using the same number of pixels as conventional equiangular cameras.

attention during image capture, it can extract novel efficiencies, particularly for bandwidth of image data. The potential for bandwidth reduction is important — Marr observed that to have foveal resolution everywhere *"...would be wasteful, unnecessary and in violation of our own experience as perceivers..."* [39]. SaccadeCam extracts the biological bandwidth advantages of attention, which impacts platforms that need perception within strict budgetary constraints, such as small robots and long-range drones. We show SaccadeCam results for distributing visual attention (using the proxy of image resolution) to improve depth estimation. **Our contributions are:**

- We define a new problem of distributing image resolution under a fixed camera bandwidth around the scene with the goal of succeeding at depth estimation (Sect. 2 and Table 2).
- We design an end-to-end network that controls resolution distribution, showing that SaccadeCam images outperform conventional distribution of resolution and can detect important objects for robot navigation, such as poles, signs and distant vehicles (Sect. 3, Table 3 and Fig. 3, Sect. 5 Fig. 4).
- We validate our method on a real hardware prototype that images multiple fovea per frame. We also present a generalized selection algorithm to extract discrete fovea from the attention mask. (Sect. 5).

Method (with few examples)	Adaptive	Test Input	Depth Recovery	Attention *during* image capture	Self/Semi/Guided
Deep Attention Mechanisms [59, 62, 29]	✓	Mono/Mono+X	✓	×	All
Compressive Imaging [14]	×	Mono/Mono+X	✓	×	All
Monocular Depth Estimation [51, 25]	×	Mono	✓	×	All
Monocular Guided Upsampling [11, 19]	×	Mono+X	✓	×	Semi/Guided
Adaptive Guided Upsampling [4, 6]	✓	Mono+X	✓	×	Guided
End-to-end Optics [9]	×	Mono	✓	×	Guided
Learned Zoom [65]	×	Mono	×	×	Guided
Adaptive Zoom [56]	✓	Mono	×	×	Self
SaccadeCam (**Ours**)	✓	Mono	✓	✓	Self

Table 1: SaccadeCam Framework vs. Other Alternatives: To our knowledge, ours is the only work that provides adaptive, monocular depth estimation by manipulating attention inside the camera, during image capture, while being self-supervised.

1.2. Related Work

Saccades, attention and related ideas have been studied in robotics and active vision for many years [1, 3, 17, 42, 12, 18, 7]. In addition, foveal designs to enable high-quality imaging are also common [43, 26, 41, 13]. Our SaccadeCam framework is different in three important ways. First, we explore rich distribution of resolution with *multiple fovea*, which has never been demonstrated before for depth estimation. Second, we apply end-to-end learning to find where to place fovea in a scene to estimate monocular depth with non-uniform spatial resolution. Finally, we demonstrate a working SaccadeCam with a microelectromechanical (MEMS) mirror that is *directly controlled by our trained networks*. We now discuss specific groups of related work, summarized in Table 1.

Attention in Deep Learning: Attention in deep learning typically involves learning the parameters of transformations of internal weights, so that the network can differentiably focus on specific regions. Recurrent attention networks, spatial transformer networks and Gaussian attention networks all learn such transformations [32, 30, 22, 28]. [44] show how to optimally select viewing tiles within a FOV for efficient video streaming in VR headsets. There are also approaches that use reinforcement learning for attention when a differentiable attention model is not available [61, 56, 57]. For example, in [56], the goal is to select from a small, fixed number of high-resolution patches to obtain better classification accuracy. In contrast, in our method, patches can be placed anywhere in the FOV, and SaccadeCam controls where patches are placed for depth estimation. In this sense, we take the goals of deep attention mechanisms *inside the camera*, changing how image resolution is distributed under a fixed camera bandwidth.

Monocular and Guided Depth Completion: Monocular depth methods have been very successful [51, 49, 36, 52, 10]. A variety of improvements on these methods by applying a "mono+X" strategy have been proposed [5, 11, 38, 37, 55, 50, 27] with an available benchmark on the KITTI dataset [55]. Upsampling has been shown with sparse depth [58], single-photon imagers [6] and flash lidar [23]. Sac-

cadeCam can be seen a first step towards physical instantiations of recent depth estimation methods that seek to self-improve imperfect measurements [55, 66, 35, 6, 46]. In contrast to these other approaches, our method is a fully passive approach that adaptively distributes resolution to enable successful monocular estimation, see Table 1.

Foveated Rendering in VR/AR: Foveation based on eye tracking is used to bypass rendering entire resolution frames in VR/AR headsets [24, 31]. [31] proposed a GAN reconstruction network that is able to take roughly 10% of an image as input and reconstruct a plausible foveated video. Rather than generating compelling viewing, we are interested in foveated imagery for depth estimation.

Compressive Sensing for Vision: Compressive signal processing uses coded optics during capture for applications such as classification [60, 15, 14]. Compressive sensing optimizes bandwidth at the cost of computing (such as L1-optimization), after image capture, to decode the measurements. Our approach is about emphasizing scene areas with new measurements during image capture, reducing bandwidth without extra computing.

Adaptive Imaging for Vision: End-to-end learning inside the camera has impacted many applications in computational cameras and computer vision. These include learning optimal structured light patterns [2], learning optimal lens parameters for monocular depth estimation [9] and HDR imaging [40], and learning optimal sensor designs [8]. SaccadeCam is different in that the optics are not fixed but foveate, enabling active, adaptive changes in imaging inside the camera. This is also what separates us from previous work that does not use learning to decide where to distribute resolution [54]. In this sense, our work is similar to adaptive LIDAR work [47, 35, 6, 46], but instead we seek to control monocular resolution for depth sensing.

2. Can Adaptive Attention Improve Depth?

Our hypothesis is that distributing pixels within a camera field-of-view can positively impact monocular depth estimation. This is only possible if models of differing bandwidths perform similarly on smooth consistent regions and

		Abs Rel	Sq Rel	RMSE	RMSE log	$\delta<1.25$	$\delta<1.25^2$	$\delta<1.25^3$
	Full Resolution (70 pixels/mm)	0.109	0.883	4.960	0.208	0.865	0.949	0.975
	Target resolution (31 pixels/mm)	0.118	0.988	5.188	0.214	0.851	0.944	0.974
(a)	Wide Angle Camera (27 pixels/mm)	0.119	0.991	5.238	0.216	0.846	0.943	0.974
	Photometric Oracle	0.116	0.941	5.134	0.213	0.851	0.945	0.975
	True Oracle	**0.114**	**0.853**	**4.850**	**0.208**	**0.857**	**0.950**	**0.976**
(b)	Wide Angle Camera (22 pixels/mm)	0.121	1.005	5.275	0.219	0.840	0.939	0.973
	Photometric Oracle	0.116	0.931	5.114	0.214	0.848	0.943	0.974
	True Oracle	**0.111**	**0.850**	**4.846**	**0.206**	**0.863**	**0.950**	**0.976**
(c)	Wide Angle Camera (15 pixels/mm)	0.128	1.067	5.507	0.228	0.824	0.934	0.971
	Photometric Oracle	0.120	0.960	5.238	0.219	0.840	0.941	0.973
	True Oracle	**0.112**	**0.847**	**4.848**	**0.206**	**0.866**	**0.951**	**0.976**

Table 2: We motivate our method with a compelling example from the KITTI dataset [20]. We want to see if increasing resolution in high error regions in the wide angle camera (WAC) lowers the overall depth error compared to the target resolution. We compare a full resolution of 70 px/mm (conventional KITTI imagery) with a target resolution of 31.30 px/mm. As expected, full resolution does far better than both target resolution and three different resolution low-res WAC images. Photometric Oracle distributes resolution based on per pixel error between WAC-trained and full resolution-trained depth networks. True Oracle distributes resolution based on the error regions between a WAC-trained depth network and ground truth LIDAR. For both True Oracle and Photometric Oracle, depth from a focused-trained depth network with focused input images is placed in the attention regions and then the error is recalculated for the result. The results show that *distributing resolution* adaptively can provide the best depth. In Sect. 3 we describe our algorithms to extract this potential of SaccadeCam.

perform differently on critical regions. We want to test this hypothesis and build learning mechanisms to distribute these pixels in a self-supervised manner, with no requirement for ground truth labels as recent work has shown [21]. Given a fixed bandwidth, the reduction of resolution in some areas frees up resolution to place onto critical regions such as pedestrians, signs, cars and foliage. In the next section, we discuss how to decide where to place the resolution and demonstrate the validity of our hypothesis. Now, we discuss the implications of our approach in Table 2.

2.1. Bandwidth

Table 2 has three baselines at different bandwidths. **We define bandwidth** as the number of angular samples across the FOV, i.e. our notion of bandwidth is identical to angular resolution. Therefore, while for practical reasons we may show images of the same spatial resolution (i.e. pixels in computer memory), they are of very different angular resolution. For all our experiments we use images with camera intrinsics from the KITTI dataset [20], from which we simulate different camera resolutions.

We simulate bandwidth by downsampling based on the scaled intrinsic matrix and then upsampling back to original resolution. This simulates a camera that, in practice, would have less resolution bandwidth over the same field of view. The three baselines in Table 2 are full resolution (70 px/mm bandwidth), target resolution (31.30 px/mm

bandwidth) and three low-resolution images that we term as wide-angle camera (WAC) bandwidth in the context of the SaccadeCam hardware in Sect. 5.

2.2. Depth from SaccadeCam Images

In our experiments we use the ground truth color images as the full resolution. The high resolution attention regions in our SaccadeCam images are also at the full resolution. We compare equiangular sampling of the target resolution with SaccadeCam images that have to be at the same bandwidth as the target resolution. *SaccadeCam images are created by fusing high resolution images into attention regions within the low-resolution WAC images.* The WAC resolution and the number of attention regions are constrained by the fact that their sum must equal the target angular resolution. While monocular images with equiangular resolutions have a variety of methods for depth estimation, these cannot be used directly on SaccadeCam images without training or fine tuning. This is because SaccadeCam images have spatially varying resolution, and in Sect. 3 we discuss how to extract depth from such monocular imagery. Now we discuss the implications of what is possible if such SaccadeCam depth estimation is solved.

2.3. Oracles

Our approach is to compare monocular depth estimation of equiangular images with SaccadeCam images, created by
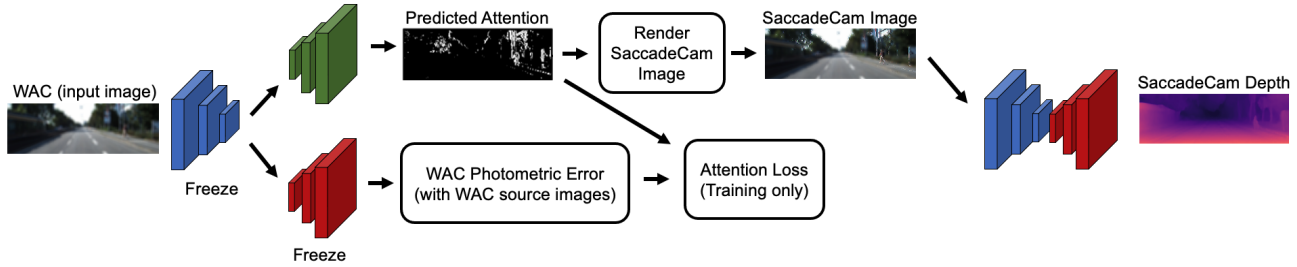
Figure 2: Our Method. We use a self-supervised setup, where the network consists of a single encoder (blue) and two decoders (red and green). Our framework takes an input image and nearby source images during training, and a single input image during testing.

unevenly distributed resolution. We design oracle experiments that determine ideal locations to distribute resolution to, and then place focused depth predictions as a perfect color-to-depth mapping in the attention regions.

For the Photometric Oracle in Table 2, the attention regions are computed based on the top N locations of the difference between the WAC depth prediction errors from a fully trained WAC network and full resolution depth prediction errors from a fully trained full resolution network using the method of [21]. We then replace the WAC depth with focused depth in the attention regions. N is the limit of available pixels left after the target resolution and WAC resolutions are determined from our camera model. We hypothesize that the focused depth errors should be lower than WAC depth errors in high resolution attention regions and similar to WAC depth errors in smooth geometrically consistent regions.

For the True Oracle in Table 2, the attention regions are computed based on the top N locations of the difference between WAC depth and ground truth LIDAR, where N is scaled according to the number of LIDAR samples versus full resolution for fair comparison. We then replace the WAC depth with focused depth in the attention regions. Therefore, if the worst depth estimates of WAC images are replaced by the corresponding depths in the same regions of full resolution images, then, as can be seen by the Table 2, depth from SaccadeCam *has the potential* to outperform state-of-the-art. Our oracle experiments support our idea that better resolution can help with depth estimation as suggested in [39, 21].

3. End-to-end Learning for Adaptive Attention

In Figure 2 we depict the complete flow for our self-supervised method. Our system consists of one encoder (blue) and two decoders (red and green). Each of these are designed for self-supervised stereo, following the method of [21]. Our method could easily be integrated with self-supervised monocular training as well, since the pose can

be estimated from multiple views of a single camera using a pose network. At test time the flow in Fig. 2 is monocular (single image), but at training time, each network takes a stereo pair.

Adaptive Attention: The attention decoder (green in Figure 2) is trained with a stereo pair of low-resolution, wide angle camera (WAC) images. The attention decoder input is the latent vector of the training depth encoder. The attention decoder then predicts per pixel attention and calculates binary cross entropy loss against the "true" binary attention mask given by the top photometric error regions calculated from the training depth network. This trains the attention mask towards 1. Our insight is that these error regions should be where additional resolution might make a difference. However, we are not strictly tied to the photometric error, as we will soon see. We then differentiably render a SaccadeCam image using the predicted attention mask, focused image, and WAC image. Here the bandwidth is given by the maximum number of samples that are possible at the highest resolution of the system. The bandwidth is a function of the target resolution and the amount of bandwidth that has already been used up by the WAC image.

SaccadeCam Rendering: Our SaccadeCam rendering module consists of alpha blending a focused image onto the WAC image using an attention mask as the blend weight. We use this to create SaccadeCam images from either a learned or oracle attention mask $\mathbf{M}$. This allows us to differentiably train our attention network end to end with a downstream monocular network,

$$I_{SaccadeCam} = \mathbf{M} \odot (I_{focused}) + (1 - \mathbf{M}) \odot (I_{WAC}). \quad (1)$$

Depth Network and Attention Regularization: The last module is the encoder-decoder pair (blue and red) that converts the SaccadeCam image into a depth. When calculating the view synthesis photometric loss [21], we compute the loss between the target SaccadeCam image and the synthesized target image that is also foveated with the same attention mask, but with the synthesized focused target image in the attention regions. The encoder and decoder used

| Target Image | SaccadeCam Image | Attention Map | Target Depth | SaccadeCam Depth |

(a)
(b)
(c)
(d)
(e)

(I) 27.11 pixel/mm bandwidth showing recovery of (a) distant car and (b)-(d) poles and (e) street sign.

(a)
(b)

(II) 31.3 pixel/mm bandwidth showing recovery of (a) signs and (b) poles.

(a)
(b)

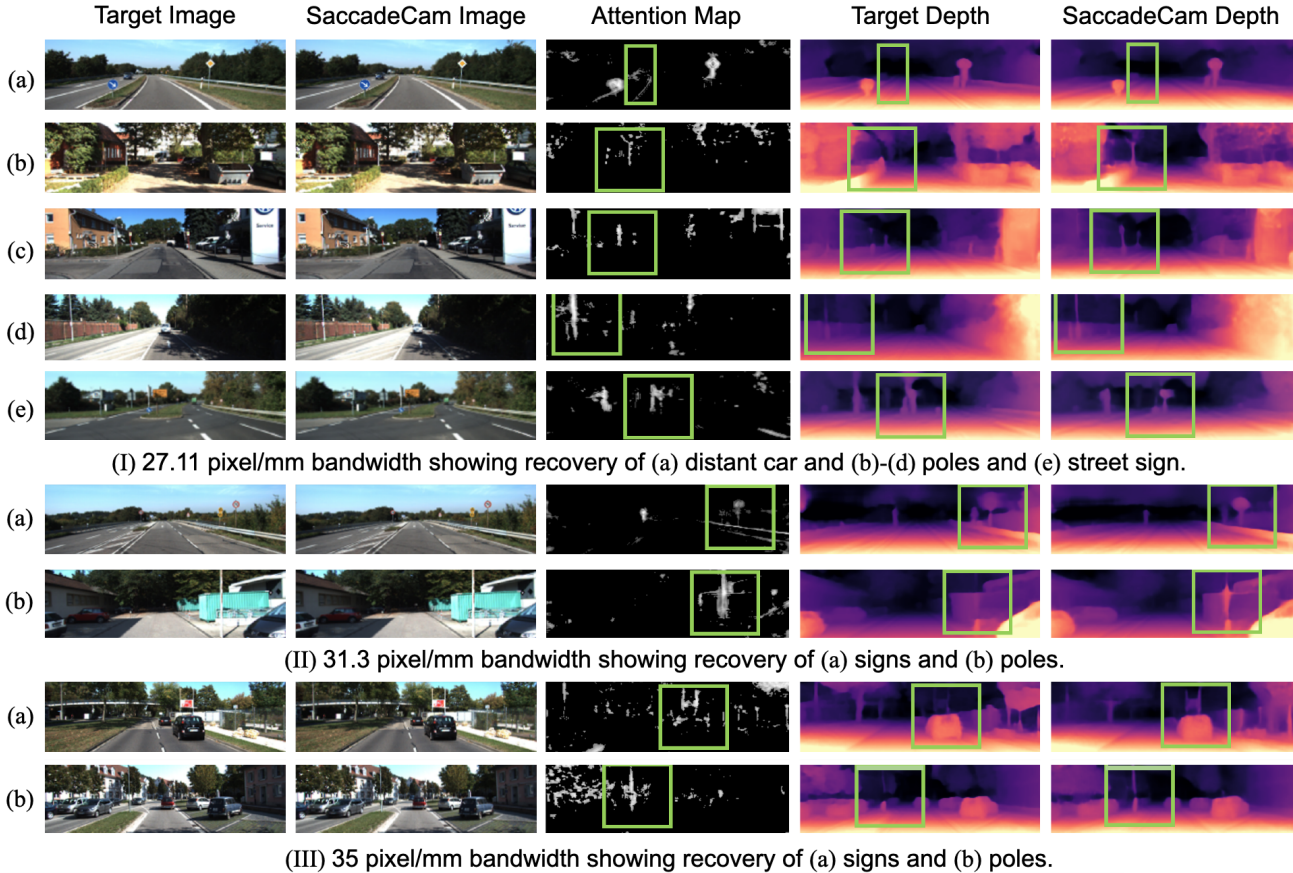(III) 35 pixel/mm bandwidth showing recovery of (a) signs and (b) poles.

Figure 3: Overview of our KITTI results. In (I-III) we show testing results from our SaccadeCam framework with progressively increasing bandwidth. Our method is particularly good at recovering thin objects such as poles or signs, that can be dangerously ignored by conventional, equiangular sampling of the scene at low resolution.

in SaccadeCam depth estimation are the same used in obtaining the WAC depth during attention estimates. During attention estimation, the gradients of the depth encoder and decoder pair are frozen. In other words, the encoder and decoder drifts towards monocular SaccadeCam image depth reconstruction, while also regularizing attention estimates. Practically, such a system is more efficient since it shares SaccadeCam features with the attention module and allows for flexible attention beyond WAC photometric errors.

Loss Terms: Our final loss is $L = \mu L_p + \lambda L_s + \alpha L_a$. L_p and L_s follow the view synthesis photometric loss and depth smoothness loss common in monocular depth estimation. We set $\mu = 1$ to avoid masking out fovea regions and $\lambda = 0.001$. L_a is the binary cross entropy loss between the predicted attention and WAC photometric error given by the SaccadeCam depth network. We freeze our depth network and set $\mu = \lambda = 0$, $\alpha = 1$ when training the attention network. We found that the attention decoder learned much quicker than the depth network (roughly 5 epochs for attention compared to roughly 20 epochs for high

bandwidth depth). We also found that an attention network trained on a single bandwidth generalizes well across different bandwidths. In an online setting, we hypothesize that infrequently updating or significantly lowering the learning rate of the attention network relative to the depth network would be beneficial.

4. Experiments

We implement our network in PyTorch on a single NVIDIA GTX 1080 Ti. Our encoder architecture is a ResNet18 and our decoder architecture is similar to [21]. All our training was initialized with ImageNet parameters. In Table 3, we show our results over a few different bandwidths. We found our SaccadeCam depth networks finished training earlier than networks trained on equiangular images based on the validation error. We train the depth networks of (a), (b), (c) for 17, 11, 2 epochs respectively and the attention networks of (a), (b), (c) for 5 epochs each. We train all equiangular resolution models for 20 epochs. *Note that*

		Fovea weighting	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta<1.25$	$\delta<1.25^2$	$\delta<1.25^3$
	Full Resolution (70 pixels/mm)		0.109	0.883	4.960	0.208	0.865	0.949	0.975
(a)	Target Resolution (35 pixels/mm)		0.117	1.001	5.144	0.213	**0.855**	0.946	0.974
	Wide Angle Camera (30 pixels/mm)		0.119	1.026	5.202	0.216	0.850	0.943	0.974
	Ours no weighting		**0.115**	**0.942**	5.087	0.209	0.853	**0.948**	0.976
	Ours fovea weighting	✓	0.116	0.950	**5.038**	**0.206**	0.852	**0.948**	**0.977**
	Color edges no weighting		0.122	0.974	5.278	0.220	0.836	0.940	0.973
	Color edges fovea weighting	✓	0.123	0.958	5.267	0.220	0.831	0.940	0.974
(b)	Target Resolution (27 pixels/mm)		**0.118**	1.013	5.209	0.215	**0.848**	0.943	0.974
	Wide Angle Camera (23 pixels/mm)		0.121	0.996	5.264	0.219	0.839	0.940	0.973
	Ours no weighting		0.121	1.003	5.192	**0.211**	0.844	**0.945**	**0.976**
	Ours fovea weighting	✓	0.119	**0.938**	**5.161**	**0.211**	0.842	0.944	**0.976**
	Color edges no weighting		0.137	1.124	5.721	0.247	0.797	0.920	0.964
	Color edges fovea weighting	✓	0.134	1.056	5.660	0.240	0.801	0.924	0.967
(c)	Target Resolution (8 pixels/mm)		0.194	2.705	7.378	0.296	0.730	0.889	0.949
	Wide Angle Camera (7 pixels/mm)		0.234	4.144	8.317	0.330	0.686	0.867	0.937
	Ours no weighting		0.167	1.516	6.815	0.270	0.743	0.900	0.958
	Ours fovea weighting	✓	**0.164**	**1.463**	**6.555**	**0.256**	**0.754**	**0.909**	**0.964**
	Color edges no weighting		0.167	1.514	6.836	0.273	0.741	0.898	0.957
	Color edges fovea weighting	✓	0.167	1.472	6.589	0.260	0.747	0.907	0.963

Table 3: SaccadeCam compared against equiangular (conventional) images. For a variety of bandwidth ratios of full resolution vs. target resolution, we show how the SaccadeCam framework (shown in Fig. 2) outperforms target resolution images with conventionally uniformly distributed resolution across the FOV.

not all bandwidths are appropriate for SaccadeCam. For example, extremely high-resolution images may not benefit from bandwidth optimization, and very low resolution images may result in extreme WAC depth errors.

We also explored weighting our loss with a weighted binary version of the predicted attention mask based on the observation that high resolution models train longer than low resolution models, this supports giving the high resolution attention region more weighting during training since the periphery is lower resolution. We train the weighted variants of (a), (b), (c) for 7, 14, 1 epochs respectively. Overall the region weighting boosts performance and speeds up training. We found at higher bandwidth Saccade-Cam data the region weighting delta must be smaller because, while the periphery is lower resolution than the high resolution attention region, it is still high enough resolution that it needs a stronger weighting to train. We weight the foveal/WAC regions of the photometric error 1.15/0.85, 1.25/0.75, 1.5/0.5 for (a), (b), and (c) respectively in Table 3.

We compare our results to monocular self-supervised depth reconstruction at the target resolution. We also compare to a color edge detector as an attention proxy. We found that edges performed well at very low resolutions, but performed poorly at higher resolutions where the fovea must be more intelligently placed to meaningfully impact performance. For our SaccadeCam networks, we first train our depth networks using the WAC photometric error as an attention proxy. We then train the attention network with the same frozen depth network using the WAC photometric error as psuedo ground truth as described in Section 3. At test time, we use the learned attention mask. We found $\geq 95\%$ overlap between the predicted attention masks and error regions on average for the test set across bandwidths, which shows the attention masks sufficiently learned to represent the error regions.

Fig. 3 shows visual results from our SaccadeCam models. Our hypothesis holds true in that we perform similar to equiangular models on smooth and geometrically consistent scene regions while outperforming equiangular models on irregular edge-case regions. Notice the SaccadeCam framework allows us to detect road signs, poles, and other distant objects such as cars that the equiangular models cannot detect.

5. SaccadeCam Hardware Prototype

Here we discuss a physical instantiation of SaccadeCam that can adaptively distribute resolution onto regions of interest based on our trained models. SaccadeCam consists of a low-resolution wide angle camera (WAC) whose field-of-view (FOV) covers the scene, and a narrow FOV telephoto camera that views reflections off a small, fast moving microelectromechanical (MEMS) mirror. These components are collectively the SaccadeCam device seen in Fig. 1.

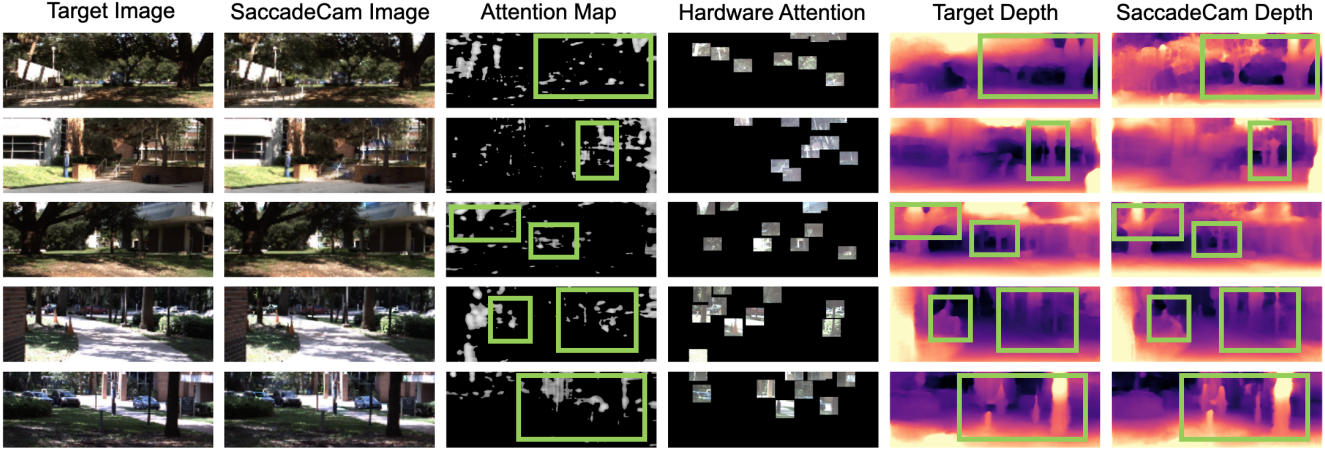Unlike many other MEMS mirror enabled devices (such

| Target Image | SaccadeCam Image | Attention Map | Hardware Attention | Target Depth | SaccadeCam Depth |

Figure 4: Results for real data captured with our SaccadeCam hardware prototype. Our trained models run on our Saccade-Cam hardware prototype to adaptively control the MEMS mirror for approximating the learned attention based on the greedy algorithm. The Hardware Attention column depicts the raw output of the MEMS images that approximate the predicted attention from our models. Our prototype is fast enough to take multiple MEMS images per WAC frame, enabling adaptive resolution distribution to several scene regions on-device at video rate.

as LIDARs [16, 53, 33]), we do not run our MEMS mirror at resonance. Instead we use a specific scan pattern, and we are able to control 5 points (i.e. 5 fovea) in the FOV at 5 Hz. This speed is reasonably fast for most objects in common scenes for depth inference. Our telephoto and WAC cameras consist of a 1.6 MP FLIR Blackfly S-U3-16S2C-CS, where the telephoto camera has a 30mm lens and the WAC camera has a 6mm lens. The telephoto camera views reflections off a 3.6mm Mirrorcle Technologies MEMS mirror with custom modifications to prevent ghosting artifacts induced from MEMS electronic packaging. Our main computer is a NVIDIA Jetson NANO, a popular embedded board with GPU and CUDA capabilities. We trace our PyTorch models to TorchScript so we can run our models on-device in C++. The Jetson NANO communicates with custom synchronization circuitry containing a Teensy 4.0 microcontroller that triggers the cameras and MEMS mirror in lockstep. The MEMS mirror is physically controlled from the Teensy through a Mirrorcle Technologies PicoAmp 5.4 X200 Digital to Analog Converter. Our hardware prototype is capable of on-device training although we leave this for future work.

5.1. Feasible Fovea from the Attention Mask

In Sect. 3 we discussed how to process the input, low-resolution WAC image to produce an attention mask across the WAC FOV, with the goal of increasing resolution in this region up to the bandwidth limit. Such an attention mask is deformable and non-convex, in the sense that there are no restrictions on optical feasibility of sensing the attention region in higher resolution, quickly.

In this section we discuss how to extract a discrete number of optically feasible saccades from the attention mask for a practical MEMS-mirror-based SaccadeCam. We also contend that it will apply to any camera that is not capable of producing programmable spatially varying deformable point spread functions (PSFs). While phase masks [64] can achieve these types of deformable attention masks, they are both slow and work best with coherent light, rather than incoherent light from a scene.

Our goal is to maximize attention mask coverage with n saccades, or mirror viewpoints. These correspond to n pairs of voltages that specify the MEMS mirror viewpoints, $\{(\theta(V(t_1)), \phi(V(t_1))), ...(\theta(V(t_n)), \phi(V(t_n)))\}$. We first tackle the problem of fixed foveal size or fovea FOV, and then we generalize such that each viewing direction i could have its own unique FOV (perhaps using a liquid lens [67]).

Greedy Attention Algorithm: The greedy algorithm requires an attention mask and a *fixed* angular fovea size ω_{fovea}. Given an attention mask defined on the FOV, $\mathbf{A}(\omega)$ where $\omega \in \omega_{fov}$, we can find the location of the maximum attention value, ω_{max} in this mask. We then follow an iterative procedure, where we capture a fovea by selecting t_1 such that the first mirror direction $(\theta(V(t_1)), \phi(V(t_1)))$ points along the central axis of the solid angle defined by ω_{max}. We then destroy attention mask information around the first maximum such that $\mathbf{A}(\omega) = 0$, where $\omega \in \omega_{fov}$ *and* $\|\omega_{max} - \omega\| \le \omega_{fovea}$. We then repeat the procedure n times for n fovea, until a set of mirror voltages are obtained $\{(\theta(V(t_1)), \phi(V(t_1))), ...(\theta(V(t_n)), \phi(V(t_n)))\}$. The proof of this method follows from the greedy selection of subsequently maximum attention values, all of which

are monotonically decreasing (i.e. ω_{max} for t_1 is less than ω_{max} at t_2 and so on). Therefore, there is no way that there exists an attention value at location ω_{missed} that is greater than the n selected values at different locations of ω_{max}, because otherwise it would have been selected for measurement at some point between t_1 and t_n. We present derivations for an advanced attention coverage algorithm based on the optical knapsack algorithm from [45] in the supplementary, although we do not implement this algorithm in hardware.

5.2. Hardware Prototype Results

We show qualitative results on real data captured with our SaccadeCam hardware prototype in Fig. 4. Our results are obtained on-device at video rate as follows. The NVIDIA Jetson NANO triggers the WAC camera and passes the WAC image through our trained attention network. Next, given a calibrated MEMS mirror, telephoto and WAC cameras, we determine the top ten pixel locations (and therefore MEMS voltages) that optimally cover the attention prediction with our greedy algorithm. The mirror is triggered and moves to a location whereby the telephoto camera is subsequently triggered to capture an image of the MEMS mirror reflection. We choose ten fovea for our hardware prototype so the previous step is repeated until ten MEMS mirror images are captured; the Hardware Attention column in Fig. 4 shows examples of the ten captured MEMS mirror images taken by the telephoto camera. We then gamma correct and blend the telephoto camera images onto the WAC image to form the SaccadeCam image. Finally, the SaccadeCam image is passed through our depth network to obtain our result.

For results with our SaccadeCam hardware prototype, we keep the target resolution bandwidth at 35 px/mm and SaccadeCam WAC bandwidth at 31 px/mm with ten fovea. This lets us use models trained on the much larger KITTI dataset. For target depth we use the 20 epoch weights of 35 px/mm target bandwidth. We finetune SaccadeCam weights for 5 epochs at 1e-7 learning rate on KITTI with patch fovea to smooth out rough square boundary edges occurring when overlaying fovea images onto the WAC image since the fovea images are square and do not perfectly approximate the learned attention.

Fig. 4 shows that our hardware prototype can qualitatively match the results seen on the KITTI test set in Fig. 3 in that SaccadeCam depth outperforms target depth in the learned attention regions thanks to the natively-high angular resolution of the telephoto camera viewing the MEMS mirror.

6. Discussion and Limitations

In this paper we provide a new framework, Saccade-Cam, for leveraging visual attention during image forma-

tion. Our key idea is to adaptively distribute resolution onto the scene, to improve depth sensing, demonstrating that our framework can perform better than equiangular distribution of pixels. We now discuss some limitations that we would like to improve in future work:

Real-time demonstrations: Our current hardware prototype allows for on-device end-to-end learning at nearly 5 Hz. We want to demonstrate dynamic scenes results soon with faster hardware.

Deformable attention masks: Our setup and theory already allow deformable attention masks, and we wish to use a liquid lens to demonstrate this.

Beyond depth estimation: The differentiable and modular nature of the SaccadeCam framework encourages integrating SaccadeCam into other existing vision applications such as semantic segmentation or pedestrian detection.

References

[1] John Aloimonos, Isaac Weiss, and Amit Bandyopadhyay. Active vision. *International journal of computer vision*, 1(4):333–356, 1988.

[2] Seung-Hwan Baek and Felix Heide. Polka lines: Learning structured illumination and reconstruction for active stereo, 2020.

[3] Ruzena Bajcsy. Active perception. *Proceedings of the IEEE*, 76(8):966–1005, 1988.

[4] Joseph R Bartels, Jian Wang, William Whittaker, Srinivasa G Narasimhan, et al. Agile depth sensing using triangulation light curtains. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7900–7908, 2019.

[5] Ramy Battrawy, René Schuster, Oliver Wasenmüller, Qing Rao, and Didier Stricker. Lidar-flow: Dense scene flow estimation from sparse lidar and stereo images. *arXiv preprint arXiv:1910.14453*, 2019.

[6] A. Bergman, D. Lindell, and G. Wetzstein. Deep adaptive lidar: End-to-end optimization of sampling and depth completion at low sampling rates. *ICCP*, 2020.

[7] Neil Bruce and John Tsotsos. Attention based on information maximization. *Journal of Vision*, 7(9):950–950, 2007.

[8] Ayan Chakrabarti. Learning sensor multiplexing design through back-propagation, 2016.

[9] Julie Chang and Gordon Wetzstein. Deep optics for monocular depth estimation and 3d object detection, 2019.

[10] Weifeng Chen, Zhao Fu, Dawei Yang, and Jia Deng. Single-image depth perception in the wild. In *Advances in neural information processing systems*, pages 730–738, 2016.

[11] Zhao Chen, Vijay Badrinarayanan, Gilad Drozdov, and Andrew Rabinovich. Estimating depth from rgb and sparse

sensing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 167–182, 2018.

[12] Donald G Dansereau, Ian Mahon, Oscar Pizarro, and Stefan B Williams. Plenoptic flow: Closed-form visual odometry for light field cameras. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4455–4462. IEEE, 2011.

[13] Trevor Darrell, Baback Moghaddam, and Alex P Pentland. Active face tracking and pose estimation in an interactive room. In *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 67–72. IEEE, 1996.

[14] Mark A Davenport, Marco F Duarte, Michael B Wakin, Jason N Laska, Dharmpal Takhar, Kevin F Kelly, and Richard G Baraniuk. The smashed filter for compressive classification and target recognition. In *Electronic Imaging 2007*, pages 64980H–64980H. International Society for Optics and Photonics, 2007.

[15] Marco F Duarte, Mark A Davenport, Dharmpal Takhar, Jason N Laska, Ting Sun, Kevin F Kelly, and Richard G Baraniuk. Single-pixel imaging via compressive sampling. *IEEE signal processing magazine*, 25(2):83–91, 2008.

[16] Thomas P Flatley. Spacecube: A family of reconfigurable hybrid on-board science data processors. 2015.

[17] Simone Frintrop and Patric Jensfelt. Attentional landmarks and active gaze control for visual slam. *IEEE Transactions on Robotics*, 24(5):1054–1065, 2008.

[18] Simone Frintrop, Erich Rome, and Henrik I Christensen. Computational visual attention systems and their cognitive foundations: A survey. *ACM Transactions on Applied Perception (TAP)*, 7(1):6, 2010.

[19] Rahul Garg, Neal Wadhwa, Sameer Ansari, and Jonathan T Barron. Learning single camera depth estimation using dual-pixels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7628–7637, 2019.

[20] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[21] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J. Brostow. Digging into self-supervised monocular depth prediction. October 2019.

[22] Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra. Draw: A recurrent neural network for image generation, 2015.

[23] Tobias Gruber, Frank Julca-Aguilar, Mario Bijelic, and Felix Heide. Gated2depth: Real-time dense lidar from gated images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1506–1516, 2019.

[24] Brian Guenter, Mark Finch, Steven Drucker, Desney Tan, and John Snyder. Foveated 3d graphics. *ACM Trans. Graph.*, 2012.

[25] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2485–2494, 2020.

[26] Hong Hua and Sheng Liu. Dual-sensor foveated imaging system. *Applied optics*, 47(3):317–327, 2008.

[27] Tak-Wai Hui, Chen Change Loy, and Xiaoou Tang. Depth map super-resolution by deep multi-scale guidance. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2016.

[28] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks, 2016.

[29] Adrian Johnston and Gustavo Carneiro. Self-supervised monocular trained depth estimation using self-attention and discrete disparity volume. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4756–4765, 2020.

[30] Samira Ebrahimi Kahou, Vincent Michalski, and Roland Memisevic. Ratm: Recurrent attentive tracking model, 2016.

[31] Anton S. Kaplanyan, Anton Sochenov, Thomas Leimkühler, Mikhail Okunev, Todd Goodall, and Gizem Rufo. Deepfovea: Neural reconstruction for foveated rendering and video compression using learned statistics of natural videos. *ACM Trans. Graph.*, 2019.

[32] Adam R. Kosiorek, Alex Bewley, and Ingmar Posner. Hierarchical attentive recurrent tracking, 2017.

[33] Krassimir T Krastev, Hendrikus WLAM Van Lierop, Herman MJ Soemers, Renatus Hendricus Maria Sanders, and Antonius Johannes Maria Nellissen. Mems scanning micromirror, Sept. 3 2013. US Patent 8,526,089.

[34] Zheng Li, Ying Wei, Yu Zhang, and Qiang Yang. Hierarchical attention transfer network for cross-domain sentiment classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[35] Chao Liu, Jinwei Gu, Kihwan Kim, Srinivasa Narasimhan, and Jan Kautz. Neural rgb-to-d sensing: Depth and uncertainty from a video camera. *arXiv preprint arXiv:1901.02571*, 2019.

[36] Miaomiao Liu, Mathieu Salzmann, and Xuming He. Discrete-continuous depth estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 716–723, 2014.

[37] Jiajun Lu and David Forsyth. Sparse depth super resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2245–2253, 2015.

[38] Fangchang Mal and Sertac Karaman. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–8. IEEE, 2018.

[39] David Marr. Vision: A computational investigation into the human representation and processing of visual information. 1982.

[40] Christopher A. Metzler, Hayato Ikoma, Yifan Peng, and Gordon Wetzstein. Deep optics for single-shot high-dynamic-range imaging, 2019.

[41] Toshiyasu Nakao and Atsushi Kashitani. Panoramic camera using a mirror rotation mechanism and a fast image mosaicing. In *Proceedings 2001 International Conference on Image Processing (Cat. No. 01CH37205)*, volume 2, pages 1045–1048. IEEE, 2001.

[42] John Oberlin and Stefanie Tellex. Time-lapse light field photography for perceiving non-lambertian scenes. In *Robotics: Science and Systems*, 2017.

[43] Kohei Okumura, Hiromasa Oku, and Masatoshi Ishikawa. High-speed gaze controller for millisecond-order pan/tilt camera. In *2011 IEEE International Conference on Robotics and Automation*, pages 6186–6191. IEEE, 2011.

[44] Cagri Ozcinar, Julián Cabrera, and Aljosa Smolic. Visual attention-aware omnidirectional video streaming using optimal tiles for virtual reality. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 9(1):217–230, 2019.

[45] Francesco Pittaluga and Sanjeev J Koppal. Privacy preserving optics for miniature vision sensors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 314–324, 2015.

[46] Francesco Pittaluga, Zaid Tasneem, Justin Folden, Brevin Tilmon, Ayan Chakrabarti, and Sanjeev J Koppal. Towards a mems-based adaptive lidar. *3DV*, 2020.

[47] Yaadhav Raaj, Siddharth Ancha, Robert Tamburo, David Held, and Srinivasa G. Narasimhan. Exploiting and refining depth distributions with triangulation light curtains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7434–7442, June 2021.

[48] Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jonathon Shlens. Standalone self-attention in vision models. *arXiv preprint arXiv:1906.05909*, 2019.

[49] Rene Ranftl, Vibhav Vineet, Qifeng Chen, and Vladlen Koltun. Dense monocular depth estimation in complex dynamic scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4058–4066, 2016.

[50] Gernot Riegler, Matthias Rüther, and Horst Bischof. Atgv-net: Accurate depth super-resolution. In *European Conference on Computer Vision*, pages 268–284. Springer, 2016.

[51] Ashutosh Saxena, Min Sun, and Andrew Y Ng. Make3d: Learning 3d scene structure from a single still image. *IEEE transactions on pattern analysis and machine intelligence*, 31(5):824–840, 2008.

[52] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer, 2012.

[53] Barry L Stann, Jeff F Dammann, Mark Del Giorno, Charles DiBerardino, Mark M Giza, Michael A Powers, and Nenad Uzunovic. Integration and demonstration of mems–scanned ladar for robotic navigation. In *Proc. SPIE*, volume 9084, page 90840J, 2014.

[54] Brevin Tilmon, Eakta Jain, Silvia Ferrari, and Sanjeev Koppal. Foveacam: A mems mirror-enabled foveating camera. In *2020 IEEE International Conference on Computational Photography (ICCP)*, pages 1–11. IEEE, 2020.

[55] Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. In *2017 International Conference on 3D Vision (3DV)*, pages 11–20. IEEE, 2017.

[56] Burak Uzkent and Stefano Ermon. Learning when and where to zoom with deep reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[57] Burak Uzkent, Christopher Yeh, and Stefano Ermon. Efficient object detection in large images using deep reinforcement learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[58] Wouter Van Gansbeke, Davy Neven, Bert De Brabandere, and Luc Van Gool. Sparse and noisy lidar completion with rgb guidance and uncertainty. In *2019 16th international conference on machine vision applications (MVA)*, pages 1–6. IEEE, 2019.

[59] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[60] Michael B Wakin, Jason N Laska, Marco F Duarte, Dror Baron, Shriram Sarvotham, Dharmpal Takhar, Kevin F Kelly, and Richard G Baraniuk. An architecture for compressive imaging. In *Image Processing, 2006 IEEE International Conference on*, pages 1273–1276. IEEE, 2006.

[61] Hanxiao Wang, Venkatesh Saligrama, Stan Sclaroff, and Vitaly Ablavsky. Cost-aware fine-grained recognition for iots based on sequential fixations, 2018.

[62] Yi Wang, Youlong Yang, and Xi Zhao. Object detection using clustering algorithm adaptive searching regions in aerial images. In *European Conference on Computer Vision*, pages 651–664. Springer, 2020.

[63] Bohan Wu, Iretiayo Akinola, Abhi Gupta, Feng Xu, Jacob Varley, David Watkins-Valls, and Peter K Allen. Generative attention learning: a "general" framework for high-performance multi-fingered grasping in clutter. *Autonomous Robots*, pages 1–20, 2020.

[64] Yicheng Wu, Vivek Boominathan, Huaijin Chen, Aswin Sankaranarayanan, and Ashok Veeraraghavan. Phasecam3d—learning phase masks for passive single view depth estimation. In *2019 IEEE International Conference on Computational Photography (ICCP)*, pages 1–12. IEEE, 2019.

[65] Xuaner Zhang, Qifeng Chen, Ren Ng, and Vladlen Koltun. Zoom to learn, learn to zoom. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3762–3770, 2019.

[66] Yinda Zhang and Thomas Funkhouser. Deep depth completion of a single rgb-d image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 175–185, 2018.

[67] Mo Zohrabi, Robert H Cormack, and Juliet T Gopinath. Wide-angle nonmechanical beam steering using liquid lenses. *Optics express*, 24(21):23798–23809, 2016.