

Capacity of Pervasive Camera Based Communication Under Perspective Distortions

Ashwin Ashok, Shubham Jain, Marco Gruteser, Narayan Mandayam, Wenjia Yuan, Kristin Dana
WINLAB, Rutgers University, 671 Route 1 South, North Brunswick, NJ, USA
Email:{aashok@winlab, shubhamj, gruteser@winlab, narayan@winlab, wenjiay, kdana@ece}.rutgers.edu

Abstract—Cameras are ubiquitous and increasingly being used not just for capturing images but also for communicating information. For example, the pervasive QR codes can be viewed as communicating a short code to camera-equipped sensors and recent research has explored using screen-to-camera communications for larger data transfers. Such communications could be particularly attractive in pervasive camera based applications, where such camera communications can reuse the existing camera hardware and also leverage from the large pixel array structure for high data-rate communication. While several prototypes have been constructed, the fundamental capacity limits of this novel communication channel in all but the simplest scenarios remains unknown. The visual medium differs from RF in that the information capacity of this channel largely depends on the perspective distortions while multipath becomes negligible. In this paper, we create a model of this communication system to allow predicting the capacity based on receiver perspective (distance and angle to the transmitter). We calibrate and validate this model through lab experiments wherein information is transmitted from a screen and received with a tablet camera. Our capacity estimates indicate that tens of Mbps is possible using a smartphone camera even when the short code on the screen images onto only 15% of the camera frame. Our estimates also indicate that there is room for a minimum of 2.5x improvement in throughput of existing screen - camera communication prototypes.

I. INTRODUCTION

The pervasive use of cameras has led to not only a diverse set of camera-based sensing applications but also to novel opportunities to use cameras to communicate information [10]. Recent efforts to standardize camera communications [3] attests the importance of using camera for communications. Camera based communication is characterized by highly directional transmission and reception along with low-multipath interference rendering it virtually interference-free. Thus, it is particularly attractive for dense congested environments where RF communication data rates are largely limited due to interference, for security applications where the directional transmissions lead to lower probability of interception or observability of signals, or for situations where the high directionality leads to improved localization of the transmitters. Camera based communication can leverage existing cameras for communicating with the ubiquitous light emitting devices. Information could be transmitted from TVs, monitors, billboards, and even projector screens. We believe, therefore, that camera-based communications can be an attractive alternative or supplement to RF wireless based communication.

Today, cameras are frequently used to read QR-codes,

however, which can be considered as a form of visual communication wherein the camera acts as a receiver. The ubiquitous use of QR codes motivates to build novel camera communication applications, where pervasive display screens could be modulated to send time-varying QR codes to be decoded by video cameras. The large pixel array elements of the screen and camera can be leveraged to send high volume of data through short time-varying 2D barcodes. For example, a user could point a camera to a desktop PC or even a smartphone screen displaying the time-varying code to download a file or perhaps a video. Recent research has further explored this direction by designing prototypes wherein time-varying 2D barcodes can be transmitted from desktop monitors [17], [20] and smartphone screens [14] to a camera receiver. While these works have designed and measured the performance of specific point solution in this space, how much room for improvement exists in these solutions or if there is any bound on performance still remains unclear.

QR-code recognition is typically limited to short distances of cm and the camera usually has to be well-aligned so that the code covers most of the camera image. The throughput is largely affected by any change in perspective (position or orientation) of the camera with the transmitter, due to perspective distortions caused by the nature of camera imaging. When the camera is far from the screen (or at a highly oblique angle) light rays from multiple transmitter elements (pixels of the screen) start to interfere on one or more camera pixels, causing inter-pixel interference (IPI), and thus reducing received signal quality and throughput.

To our knowledge, only few projects have begun to investigate information capacity limits of camera communication using time-varying 2D barcodes. Hranilovic et.al. [17] analyze the capacity and prototype a screen - camera system where a CCD camera was placed at a fixed distance from a laptop screen, but does not account for the interference between pixels and the dependence on perspective. The model in [10] can be considered as a simplified case of screen - camera channel where the transmitter and receiver always remain aligned, while ignoring the quantization effects of real camera receivers.

In this paper, we develop a model for the information capacity of screen-camera communication that accounts for perspective dependent (position and orientation) distortions that dominate this channel. The model incorporates projection theory from the computer vision domain into a Shannon

capacity formulation. Specifically, our contributions in this paper are:

- A screen-channel model that accounts for perspective distortions and realities of camera receivers such as quantization limitations. We model the camera perspective distortion in detail using camera imaging theory.
- Experimental calibration and validation of the model through extensive lab measurements using a screen transmitter and a tablet camera receiver. The study parameters include distance, angle, granularity or blocksize of the code (number of pixels per transmit bit).
- Estimation of capacity for screen-camera communication. We compute capacity experimentally by measuring channel and signal quality metrics, such as bandwidth and signal-to-interference-noise ratio, and validate by substituting the same into the derived analytical capacity expression.
- Compare capacity estimate with throughput of existing screen-camera communication prototypes and show that there is a large room for improvement.

II. BACKGROUND ON CAMERA COMMUNICATION

Camera based communication is a class of visible light communications (VLC) [8], where information is modulated through light transmitted from optical emitters such as LEDs and LCDs, and received by photo-receptor elements at the receiver (in a camera the image sensor pixels are the photo-receptors). The inherent 2D spatial array structure of the image sensor pixels can be leveraged to create a multi-input-multi-output (MIMO) channel by using arrays of optical emitter elements to transmit information through a concept called visual MIMO [10]. In this regard, the array of LEDs in lighting arrays and commercial display devices, LCD pixels in display screen, projector screens, or printed material¹ qualify as potential transmitters in camera based communications.

A camera channel is analogous to a RF MIMO channel where each pixel element of the camera acts as a receiving antenna and the light emitting elements as the transmit antennas. In RF MIMO, the signal quality at each receive antenna element is a function of the path-loss in the channel, multipath fading, and the interference from other transmit antennas — also called co-channel interference [7]. A camera channel has negligible multipath fading but experiences visual distortions in the image due to path-loss in light energy, and interference from other light emitting elements. Unlike RF channels, the camera channel is only partially random (where randomness is mainly due to background noise) and can potentially be modeled using classical camera imaging theory.

Noise in camera channels typically have additive-white-Gaussian-noise (AWGN) characteristic [23]. Noise in camera pixels manifests as *noise current* generated due to the photons from the ambient lighting and those generated from the imaging circuit [19]. The noise current in a pixel can be considered

signal independent when the ambient lighting is sufficiently high compared to the transmit signal, such as in office rooms or outdoors [21]. At the output of a camera, the noise current in each camera pixel is a quantized quantity and manifests as fluctuations in the intensity (digital value of the sensor output) of that pixel; the noise energy accumulated in each pixel can be quantified using the mean variance in the pixel intensity. Noise from background lighting is typically considered to be uniform over the image sensor (photoreceptor), and quantified through the AWGN noise-variance σ_n^2 [21], [23].

Considering the deterministic nature of perspective distortions and the AWGN channel, capacity (measured in bits/sec) of camera based communication can be expressed using Shannon Capacity formula as,

$$C = W_{fps}(W_s \log_2(1 + SINR)) \quad (1)$$

where $SINR$ represents the signal-to-interference-noise ratio per pixel, W_{fps} is the camera-frame rate or the receiver sampling rate in frames-per-second. W_s is the spatial-bandwidth, which denotes the number of information carrying pixels per camera image frame. The spatial bandwidth is equivalent to the number of orthogonal channels in an RF MIMO system. In the rest of the paper, we will use the terms *screen transmitter* and *screen* interchangeably to refer to the variety of light emitting array equipped display devices that can be used as transmitters in camera communication, and the term *image* to refer to the camera sampled image.

III. SCREEN - CAMERA CHANNEL

In screen - camera communication, information is modulated in the light intensity of the pixel elements of a screen transmitter that are received and decoded from the camera image pixel intensity at the receiver. The pixel intensity in a camera image is a digital quantity² that is proportional to the amount of photon current generated on the pixel from the light energy accumulated over its area (smaller the pixel area lesser the light intensity it accumulates). When the light emitting screen pixel is at the focus of the camera lens (and hence the camera pixel) all the light rays from the screen pixel are focused onto the camera pixel and thus incurring no loss of energy on the pixel. But when the screen pixel is perturbed (in position or orientation) from the focus of the camera, due to the finite aperture size of the camera lens, not all light rays converge on the camera pixel resulting in reduced accumulated energy and hence a smaller pixel intensity value. The loss in the received light intensity on a camera pixel results in the visual deformation in size or shape of the imaged screen pixel; an effect that is termed as perspective distortion.

A. Perspective Distortions

Perspective distortions are caused due to the nature of the camera imaging mechanism and manifest as deformation in size and shape of the captured object (the light emitting screen pixel) on the image, resulting in visual compression or

¹Barcodes such as QR codes printed on papers qualify as time-invarying message

²cameras pixels typically have 8 bit monochromatic depth where the values span 0 (dark)-to-255 (bright)

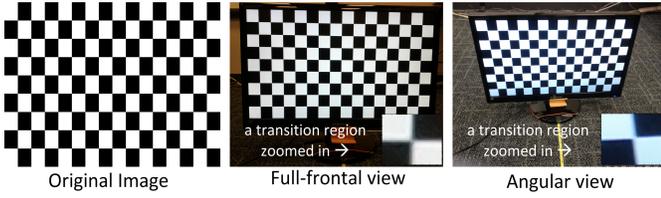


Fig. 1. Illustration of perspective distortion in screen-camera channel. Imaged screen pixels are blurry, and reduced in size in full-frontal view and also in shape in angular view.

magnification of the object's projection on the image. When the screen is at an out-of-focus distance from the camera lens (or at an oblique angle), these distortions become prominent and lead to interference between adjacent screen pixels on the camera image, what we term as inter-pixel interference or IPI. The combined effect of background noise and IPI degrades the received signal quality and hence reduces information capacity in camera channels.

For example, let us consider that blocks of screen pixels are illuminated by a chessboard pattern and imaged by a camera as shown in Figure 1. We can observe that perspective distortions cause the screen pixels to deform in size when the screen is not at the focus of the camera, and in shape when it is not frontally aligned (viewed at an angle) with the camera. If the screen pixel was at the focus, and assuming the screen and camera have the same resolution, it's image on the camera should be the same size as one pixel. But in reality, the light rays from the screen pixel may not end exactly on camera pixel boundaries and there is some area surrounding it that accumulates interference. This area of misalignment and the size (or shape) of the imaged screen pixel will be perspective (position and orientation) dependent and accounts for distortion due to perspective scaling of the pixel area.

We can also observe from Fig. 1 that the imaged screen pixels are blurry, especially at the transition regions between white and black blocks. This blur effect is attributed to the camera lens and more formally termed as lens-blur or point-spread function (PSF) in computer vision theory [16]. The PSF represents the response of an imaging system to a point source. PSF causes the received light energy to spread to areas outside the pixel, where the amount of spread depends on the type of lens being used. Lens-blur can be understood as a low-pass filtering phenomenon that distorts the high-frequency components in the image, such as edges and high contrast regions [13]. In the screen - camera channel this translates to distorting the pixels at the transition regions between brighter (high intensity) and darker (low intensity) pixels, and leads to interference (IPI) between neighboring pixels as seen in Fig. 1. Since the area and the maximum energy that can be sampled in each camera pixel is finite (a camera pixel output is a digital quantity), IPI leads to an effective reduction in signal energy per pixel.

Unlike fading in RF wireless channels, perspective distortions are deterministic, and can be modeled using camera imaging theory as exercised in this paper. As shown in

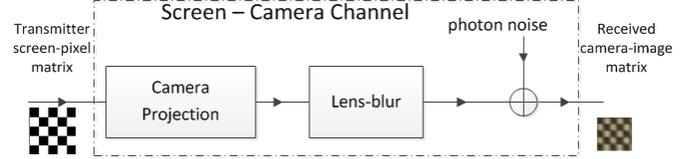


Fig. 2. Screen - Camera Channel Model

Fig. 2, we will treat the screen-camera channel distortions as a composite effect of signal energy reduction due to perspective scaling of pixel area owing to camera projection, signal energy reduction due to lens-blur, and background photon noise.

Perspective Distortion Factor:

Let us consider that the signal energy on each pixel is weighted by perspective distortion factor α that represents the effective area scaling (down) due to perspective scaling and lens-blur in the camera imaging process. Based on its definition this factor takes values such that $0 \leq \alpha \leq 1$, where $\alpha = 1$ indicates that the screen pixel is at the focus of the camera and also incurs no signal reduction due to lens-blur, and $\alpha = 0$ indicates that no part of the screen-pixel gets imaged on the camera pixel.

As discussed earlier, lens-blur (PSF) causes the signal energy to leak outside the area of a single pixel (on both X and Y dimensions of the pixel). Camera lens-blur (PSF) can be approximately modeled as a 2D gaussian function [6], [16], where the amount of spread is quantified using its variance σ_{blur}^2 (a large variance indicates more blur³). In our model we account for lens-blur distortion using the factor $\alpha_b = 4\sigma_{blur}^2$ – to account for the spread on the four sides of square pixel. If the area of a pixel is s_{cam}^2 (where s_{cam} is the side length of the camera pixel), then the effective signal energy on that pixel will be proportional to $s_{cam}^2 \frac{1}{1+\alpha_b}$. We treat this signal energy reduction is equivalent to the effective reduction in pixel area over which the signal accumulates.

Let α_p represent the perspective scaling of the area of an imaged screen pixel when perturbed from camera focus. We model this perspective scaling factor and derive a general expression for α_p in Appendix A using camera projection theory [15], that uses the camera projection matrix which maps the location of the screen pixels from the world coordinate frame to the camera coordinate system. As can be seen from equation (17) α_p is a function of the distance and angle between the screen and camera. In the simplest case, where the screen and camera are perfectly aligned at distance d , this factor can be expressed as,

$$\alpha_p = \left(\frac{f_{cam} s_t}{s_{cam} d} \right)^2 \quad (2)$$

where f_{cam} , s_t are the focal length of the camera and side-length of the screen pixel, respectively.

We can observe from equation (2) that, $\alpha_p = 1$ when the camera is at the focus ($d = f_{cam}$) and if $s_{cam} = s_t$. But in

³For an ideal pin-hole camera energy spread over a pixel would be uniform and hence σ_{blur}^2 is infinitesimally small

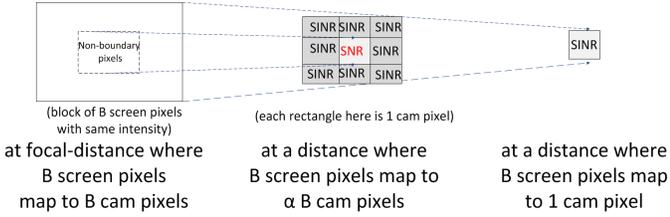


Fig. 3. Illustration of interference between pixel-blocks due to perspective distortion for SINR computation

reality, the physical size of a screen and camera pixel may not be the same. In our model we treat the camera focal point as the distance $d_f = \frac{f_{cam}s_t}{s_{cam}}$, what we call *focal-distance*. In reality, we envision that communication distances in screen-camera communication applications are much larger than d_f (for example, $d_f = 39cm$ for the test LCD screen-tablet camera system we evaluate in section V).

If α denotes the average distortion in each pixel of the camera image, we express α as the effective pixel area reduction due to perspective scaling factor α_p on the reduced pixel area due to lens-distortion $\alpha_b = 4\sigma_{blur}^2$,

$$\alpha = \alpha_p \times \frac{1}{(1 + \alpha_b)} \quad (3)$$

B. Signal-to-Interference Noise Ratio

We quantify the quality of the signal at the camera receiver in the screen-camera channel using the average SINR per pixel,

$$SINR_\alpha = \frac{\alpha P_{avg}^2}{(1 - \alpha)P_{avg}^2 + \sigma_n^2} \quad (4)$$

where, P_{avg} denotes the average transmit pixel intensity (for example, a screen-camera system using black (0) and white (255) intensities for transmission will have $P_{avg} = 127.5$). By using the digital value of the average signal P_{avg} , instead of its analog equivalent which is the pixel photon-current, our model accounts for the quantization limitations in cameras. The $1 - \alpha$ term in equation (4) quantifies the fraction of the pixel area affected by interference. σ_n^2 denotes average AWGN noise energy in each pixel, and since noise uniformly affects the entire area the pixel it does not depend on the scaling factor α .

Pixel blocks: A small value of α means more screen pixels interfere on one camera pixel. In reality, screen pixels are very closely spaced (fraction of a mm), and so, IPI will be inevitable even at short distances (since α_p is very small as s_t is very small) resulting in very low SINR. A potential solution is to leverage the MIMO structure of the screen-camera channel, by grouping multiple screen pixels in a block to transmit with same intensity and combine those imaged pixels from the camera image to improve SINR. This process is similar to the diversity combining used in RF MIMO. Though, using the entire screen to transmit just one intensity value is wasteful; a better technique is to divide the screen into a set of blocks

where intensities (bits) are multiplexed over each set and each subset block has a group of pixels transmitting with the same intensity.

The perspective scaling factor α_p derivation (in Appendix A) accounts for the misalignment between screen and camera pixels as it uses camera projection theory that maps the screen pixels to camera coordinates. Error due to misalignment will project as deviation in the distortion factors for each pixel. By taking an average value over the camera image this error becomes negligible. Though, one can assume some vibrations on the pixels, especially when the camera is not stable⁴, which means that the area of misalignment can keep changing with perspective.

The impact of misalignments, and also lens-blur, will become smaller as one block covers more pixels on the camera and only affect pixels near the boundary as shown in Fig.3. This is because, all the neighboring pixels for non-boundary block is a pixel from the same parent block transmitting the same intensity. This implies that ‘interference’ for a non-boundary pixel is negligible, even for a non-zero blur or pixel misalignment, as the same signal adds-up on the pixel enhancing signal energy of that pixel, in which case the SINR of that pixel converges to the average signal-to-noise ratio. As a convention in our model, we treat a pixel block as a boundary block if it is not all surrounded by blocks with same intensity.

In general, the expression for the average SINR per imaged block in a screen-camera channel, using B pixel square blocks of a screen can be given as,

$$\begin{aligned} SINR_{blk}(\alpha, B) &= [\gamma_1 SINR_\alpha + \gamma_2 SNR_\alpha], \forall \alpha B > 4 \\ &= SINR_\alpha \quad \forall \alpha B \leq 4 \end{aligned} \quad (5)$$

where $SINR_\alpha$ is from equation 4, $SNR_\alpha = \frac{\alpha P_{avg}}{\sigma_n^2}$, and the coefficients $\gamma_1 = 4(\sqrt{\alpha B} - 1)$ and $\gamma_2 = (\sqrt{\alpha B} - 2)^2$ represent the number of boundary-blocks and non-boundary blocks, respectively. We consider that $\min B = 4$ (i.e. 2×2 pixels), and $\alpha B \leq 4$ indicates that each B screen-pixel block projects onto a maximum of 1 camera pixel area and $\alpha B > 4$ indicates that the block projects onto multiple camera pixels.

IV. CAPACITY UNDER PERSPECTIVE DISTORTIONS

Recalling the capacity expression from equation. (1), we can express the capacity of screen-camera communication in bits/sec as,

$$C_{cam}(\alpha) = \frac{W_{fps}}{2} \alpha ||R_{cam}|| \log_2(1 + SINR_\alpha) \quad (6)$$

where $SINR_\alpha$ is the signal-to-interference noise ratio from equation (4), $||R_{cam}||$ denotes resolution of the camera and W_{fps} denotes the frame-rate of the camera in frames-per-second. The factor of half in the frame-rate is attributed to the synchronization mismatch between screen-pixel modulation rate and camera sampling rate or frame-rate. To avoid aliasing

⁴definitely true for handheld cameras, but also applies for many more stationary scenarios

between two successive sampled samples of the screen-pixel by the camera, the maximum sampling rate is halved - according to Nyquist sampling theory.

The term $\alpha||R_{cam}||$ represents the total number of camera pixels that contain the image of the screen pixels, and is essentially the spatial-bandwidth term W_s in equation 1. This is very different from RF MIMO, where, all the receiver antennas can potentially receive the signal, independent of distance between the transmitter and receiver. In a camera receiver, due to its LOS nature, the signal from each transmit element is always limited to a finite number of, but never all, receive elements.

Throughput when using pixel-blocks: As discussed earlier, using multiple screen pixels to transmit the same bit (same pixel intensity) and diversity combining them at the camera image will improve SINR. But using multiple screen pixels for a single bit will effectively reduce the total number of parallel channels (number of unique bits) available between the screen and camera pixels, and hence reducing the throughput gains achievable by multiplexing bits on the parallel channels. This nature of tradeoff between the multiplexing gains on throughput and diversity gains on SINR is analogous to the classical multiplexing-diversity tradeoff in RF MIMO [11], since the screen-camera channel MIMO structure is similar to a RF MIMO channel with a finite number of transmit and receive antennas.

The capacity in equation (6) represents the upper bound on the throughput achievable in screen-camera communication. By grouping pixels into blocks the SINR improves and number of bit errors are reduced, but at the cost of a reduced throughput. If $T_{blk}(\alpha, B)$ represents the maximum throughput of screen-camera communication for distortion α , and block-size B , then

$$T_{blk}(\alpha, B) = \frac{W_{fps}}{k} \left(\frac{\alpha ||R_{cam}||}{B} \right) \log_2(1 + SINR_{blk}(\alpha, B)), \quad (7)$$

where $\frac{\alpha ||R_{cam}||}{B}$ represents the number of parallel channels for multiplexing, and $SINR_{blk}(\alpha, B)$ is from equation (5).

The factor k on the frame-rate of the camera is attributed to the practical synchronization limitations in camera ($k \geq 2$ depending on the camera). The factor k implies that a minimum of k samples of the camera pixel are required for reliable decoding. In practice, to minimize detection and decoding errors, the camera frame-rate has to be synchronized with the modulation rate of pixel intensities on the screen as well as the refresh rate of the screen (typically 120Hz). Synchronization of cameras for communication is challenging due to the jittery nature (owing to software limitations and hardware design errors) of the frame-sampling using CMOS sensors that are widely used in mobile devices today.

V. EXPERIMENTAL CALIBRATION AND VALIDATION

In this section we describe the experiments we conducted to validate our screen-camera channel model and estimate the capacity of screen-camera communication. In particular, we

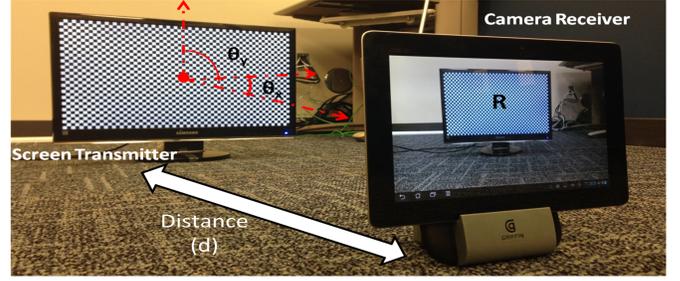


Fig. 4. Experiment setup showing LCD screen displaying black and white blocks of $B = 60 \times 60$ pixels each

show the effect of perspective (position and orientation) and design parameters such as screen pixel block-size on data-rates of camera communication. We compare our estimates of the maximum throughput with those of existing prototypes.

A. Objectives

The experiments discussed in this section aim at four key objectives;

- 1) To estimate capacity using signal-to-interference noise ratio and α measurements and compare with analytical channel capacity from equation (6).
- 2) To measure the perspective distortion factor α and validate it with the analytical model from equation (15).
- 3) To measure the SINR and validate it with the analytical model from equation (4).
- 4) To determine the noise variance σ_n^2 through measurements.

B. General Experiment Methodology

The experiment setup consisted of a 21.5inch Samsung LCD screen monitor of resolution $R_s = 1920 \times 1080$ pixels that served as the screen-transmitter and a 8MP camera of a ASUS Transformer Prime tablet (that ran Android OS version 4.1) that served as the camera receiver. The camera was operated at a resolution of $R_{cam} = 1920 \times 1080$ and with no image compression. Exposure setting and white-balancing on the camera were set to auto (default setting in android devices), though the lighting conditions were retained same for all our experiments. All our measurements were taken indoors in a lab-conference room setting equipped with fluorescent ceiling lighting⁵. We fixed the screen and tablet onto stands so as to ensure the least amount of error in the measurement of distance and angle⁶ between the tablet and camera image planes. Our raw dataset for analysis consisted of image snapshots of the screen captured by the tablet's camera at resolution of R_{cam} pixels using a standard image capture android application.

Measurements were taken by capturing snapshots of the screen displaying a chessboard pattern (blocks of B pixels each) on the LCD screen as shown in Fig. 4). The pixel-intensity of a white block was set to 255 and the black at 25

⁵typical office room lighting is 350-500 lux which is considered as 'bright'

⁶we cross-checked the angles using the rotation matrix determined from accelerometer and gyroscope readings

on the screen (the average intensity $P_{avg} = 140$). The image datasets consisted of 100 snapshots of the screen displaying the chessboard pattern, with the ceiling lights ON (an another dataset with lights OFF), at a set of distances, angles, and block-sizes. We changed the angle by only rotating the screen with respect to the X axis, since individual distortions on either X or Y axis can be considered symmetrical.

Camera Calibration: We obtained the the camera parameters, such as the focal length, pixel-side length, etc., through camera calibration procedure using the Caltech calibration toolbox [4]. Using the calibration parameters we determined that focal-distance to be $d_f = 39cm$. We also measured the blur variance σ_{blur}^2 by experimentally measuring the PSF of the tablet camera. The experiment involved emulating a point light source by illuminating one pixel on the LCD screen, and capturing its image from a distance of d_f (so as to minimize any perspective scaling). Our results indicated that a Gaussian curve with a variance $\sigma_{blur}^2 = 0.25$ was the best fit to our measurements on each dimension. Table II summarizes the list of channel measurements from our experiments, along with measured screen and camera parameters.

C. Channel Capacity

We evaluate the *measured capacity* of screen-camera channel by substituting the measured values of $SINR_\alpha$, perspective distortion factor α (and α_p , σ_{blur}), and noise variance σ_n^2 in equation. (6). We evaluate capacity in bits per camera pixel as $C_{camera}(\alpha) = \frac{C_{cam}(\alpha)}{0.5W_s||R_{cam}||}$.

The measurement procedure for α , $SINR_\alpha$, σ_n^2 are explained in detail in sections V-D, V-E, and V-F respectively.

Capacity v/s Perspective distortion factor α :

We plot the measured capacity in bits/camera-pixels for different perspective distortion factor values in Fig. 6 (a). The distortion factor α on the x-axis is comprehensive of the α s obtained for each distance and angle combination. Fig. 6 (a) shows the our estimate from measurements fit well with the model (maximum error margin of 3%).

We can observe that, about 1bit/camera pixel is achievable even when the screen is perspective scaled onto only 15% on each dimension ($\alpha = 2\%$) of the camera image. For the LCD-tablet system we used, this translates to a distance of 2.6m ($\frac{d_f}{0.15}$). At a sampling rate of 30fps⁷ and at a resolution of 1920×1080 , a data-rate of 31Mbps is achievable from an average-sized LCD monitor and a tablet camera. Assuming all parameters are the same, except the size of the screen is doubled, the same data-rate can be achieved at twice the range.

Throughput with Block-size: We plot the screen-camera communication throughput in bits-per-frame ($\frac{T_{blk}(\alpha, B)}{kW_{fps}}$) versus α for different block sizes B in Fig. 5 (b), where $T_{blk}(\alpha, B)$ is from equation (7). The steep fall-off on capacity (see $B = 15^2$ and 30^2) at small value of α is attributed to the low SINR at those perspectives. The trend in Fig. 5 (b) indicates that, while a small blocksize may yield considerable throughput at

COBRA	PixNet-C	PixNet	QR-P
4.5x	3x	2.5x	7x

TABLE I
RATIO OF CAPACITY OVER EXISTING PROTOTYPE'S THROUGHPUT (3X INDICATES THE EXISTING PROTOTYPE IS 1/3RD OF CAPACITY)

small distance (or angles) it may be better to switch to a larger blocksize at farther distance (or angle), and if such adaptations are not possible then it is judicious to use an optimal blocksize. For example, for the blocksizes in Fig. 5 (b) $B = 30^2$ looks close to optimal.

Comparison with Prototypes: We compare our throughput estimate ($T_{blk}(\alpha, B)$) with existing prototypes of screen-camera communication. PixNet [20], modulates data using OFDM and adds reed-solomon coding, and display using black and white blocks of size 84×84 . PixNet uses a 30inch LCD screen as the transmitter and 6MP CCD camera at the receiver, and tested up-to a maximum distance of 14m. The authors also reported the throughput from their implementation of QR codes, which we will call QR-P.

COBRA [14] uses color barcodes to communicate between smartphone screen and camera, with maximum test distance of 22cm with a blocksize of 6×6 . The authors also implemented a smartphone version of PixNet which we will call PixNet-C. In table I, we report the ratio of throughput $T(B)$ from equation 7 to the throughput of the these prototypes for the same parameter settings of blocksize and α in their system. Our estimates indicate that there is room for atleast 2.5x improvement in throughput when compared to capacity.

D. Perspective Distortion Factor

The objective of this experiment was to determine the perspective distortion factor α from our measurements to compute the *measured capacity*. Since α quantifies the relative area occupancy of the screen in the camera image, we measured the average distortion factor as,

$$\alpha_m = \frac{||R||}{||R_{cam}||} \frac{1}{(1 + 4\sigma_{blur}^2)} \quad (8)$$

where $||R||$ represents to the total number of camera pixels that correspond to the imaged screen pixels, and R_{cam} is the resolution of the camera. In Fig. 6(b) and 6(c) we plot α_m with angles and distance, respectively. As can be seen from these plots the measured spatial-bandwidth also fits well with the model (maximum error margin of 1.5%). The α reported here is the perspective distortion factor for our LCD - tablet channel. The distance and angle at which $\alpha = 0$ in these plots can be construed as the communication range of a system only with the same screen and camera parameters. For a screen with 10x the size (for example, a billboard [2]) the range is close to 10x that of our experimental system. Since α is the relative occupancy of the entire screen's image on the camera image, it does not depend on the block-size being used for communication.

⁷Typical frame-rate on smartphone/tablet cameras is 30fps. iPhone 5S has a 120fps capability [5]

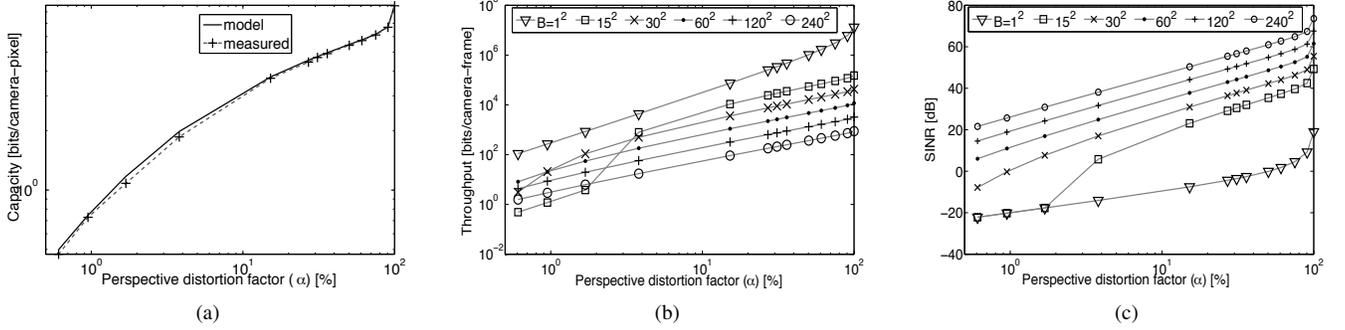


Fig. 5. (a) Capacity in bits/camera pixel ($C_{camera\text{-}pixel}(\alpha)$) for different perspective scaling (α) of screen image on camera (b) Throughput in bits/frame v/s α for different block sizes (1 frame = R_{cam} pixels, $B = 15^2$ means 15×15 pixel block on screen) (c) SINR per block v/s α for different block sizes B

E. Signal-to-Interference Noise Ratio

We use the measured signal-to-interference noise ratio $SINR_{\alpha meas}$ to compute the *measured* capacity. Let $W_{iON}(x, y)$ and $W_{iOFF}(x, y)$ represent the intensity of a pixel from a white block at location (x, y) on the camera image where the lights were ON and OFF respectively, and i ($i = 1, 2 \dots 100$) being the index of the image in the dataset (similarly let $B_{iON}(x, y)$ and $B_{iOFF}(x, y)$ represent a pixel intensity from a black block). Let $SINR_W$ denote the signal to interference noise ratio for the white pixel and $SINR_B$ for the black. We determine the measured $SINR_{\alpha meas}$ as,

$$SINR_{\alpha meas} = \frac{1}{2} \left(\sum \frac{SINR_W}{\|W\|} + \frac{SINR_B}{\|B\|} \right) \quad (9)$$

$$SINR_W = \gamma_{1m} \frac{s(W)}{k(B) + n(W)} + \gamma_{2m} \frac{s(W)}{n(W)}$$

$$SINR_B = \gamma_{1m} \frac{s(B)}{k(W) + n(B)} + \gamma_{2m} \frac{s(B)}{n(B)}$$

$$s(W) = \frac{1}{100} \sum_{i=1}^{100} \sum_{x,y} (\alpha_m W_{iON}(x, y))^2$$

$$k(B) = \frac{1}{100} \sum_{i=1}^{100} \sum_{x',y'} (1 - \alpha_m) (B_{iOFF}(x', y'))^2$$

$$n(W) = \frac{1}{100} \sum_{i=1}^{100} \sum_{x,y} (W_{iON}(x, y) - W_{iOFF}(x, y))^2 \quad (10)$$

where $(x', y') \neq (x, y)$, $\|W\|$ and $\|B\|$ represent the total number of white and black blocks respectively. γ_{1m} and γ_{2m} represent the measured number of pixels on the boundary and non-boundary blocks of the imaged block respectively.

We plot $SINR_{\alpha meas}$ (from equation. (10)) versus α with the analytical $SINR_{\alpha}$ from equation (5) in Fig. 6 (a). We can observe from Fig. 6 that our measurements are in close agreement with our model (maximum error margin of 1.5dB).

We plot the per-block measured SINR ($SINR_{blk}(\alpha, B)$) using $SINR_{\alpha meas}$ versus α for different block-sizes B in Fig. 5 (c). We can infer that larger the block size higher

is the the per-block SINR. We can also observe that for a blocksize $B = 1$, though it provides large number of parallel channels for multiplexing, the signal energy on each channel is much lower than the noise level, even for medium values of α . A low SINR indicates leads to large detection errors if no additional receiver processing is performed. While forward-error correction codes can help it adds additional overhead reducing the net throughput. But the gains due to multiplexing may be large enough so as to provide a net throughput gain. The choice on the size of blocks becomes a primary design parameter as it effects system performance.

F. Noise Measurement

The objective of this experiment was to measure noise power, to aid capacity computation for our test channel. The dataset for this experiment consisted of 200 continuous camera snapshots of the screen at 2m (and perfect alignment), displaying gray-level intensities from 0-255 in steps of 5 (total 52 sets). Based on our measurements we realized that the intensity mapping between screen and camera can be linear approximated (as shown in Fig. 7) and can be numerically expressed as $g(x) = 0.6481x + 10.06$ where $x = 0, 1, \dots 255$, and the constant 10.06 accounts for the deterministic DC noise in the pixel. The bias-factor of 25 in the measurements (shown in Fig. 7) is from the screen backlighting (the screen has a minimum brightness even when switched off). The factor 0.6481 can be treated as the path loss factor analogous to RF. As mentioned earlier, the AWGN noise from the background manifests as the temporal variance in the pixel intensity. We compute the noise energy per pixel in our LCD screen- tablet camera channel, using the mean-variance ($\hat{var}(g(x))$: averaged over 52 samples) of the intensity mapping between the screen's actual intensity and the measured intensity on the camera pixel as, $\sigma_n^2 = 10.06^2 + \hat{var}(g(x)) = 101.28$.

VI. RELATED WORK

Camera based communication is an example of visual MIMO communication [10] where camera is used as a receiver for information transmitted from arrays of light emitting elements. In [10] the authors estimate the capacity of a camera channel treating the transmitter light emitting array and the

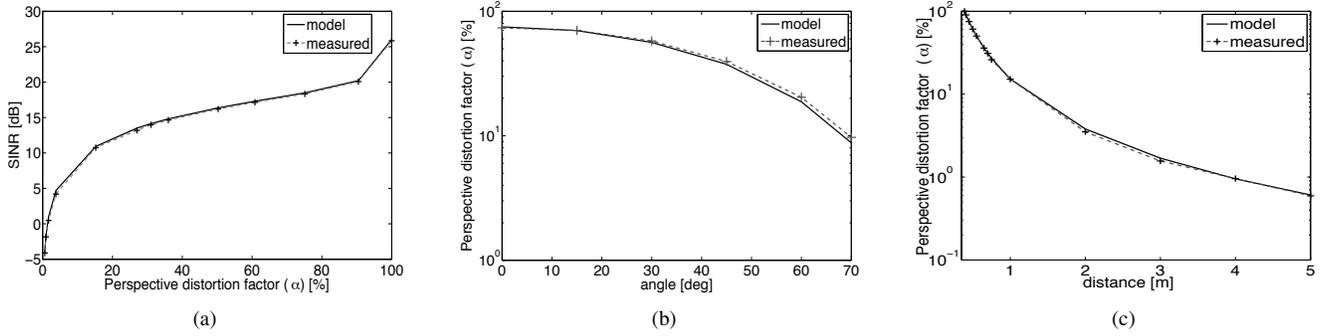


Fig. 6. (a) SINR for different perspective scaling (α) of screen image on camera (b) Perspective distortion α v/s angle between screen and camera (c) Perspective distortion factor α v/s distance between screen and camera

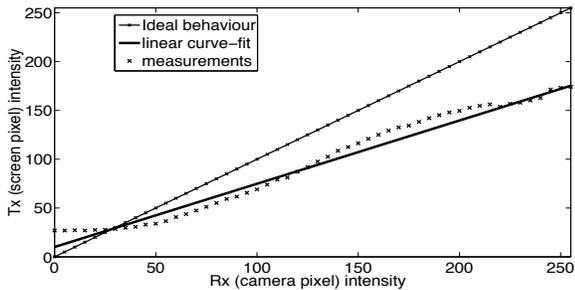


Fig. 7. Screen-Camera mapping

Parameter	Value
Cam pixel side-length s_{cam} [μm]	65
Cam focal length f_{cam} [$\times s_{cam}$]	1573
Screen pixel side-length s_t [mm]	0.248
Principal point (o_x, o_y)	(960.1, 539.2)
Noise-variance σ_n^2	101.28
Lens-blur variance σ_{blur}^2 [$\times s_{cam}^2$]	0.25
$\ R_s\ $ ($=\ R_{cam}\ $) [pixels]	1920×1080
Focal-distance d_f [m]	0.39

TABLE II
TABLE OF SCREEN, CAMERA AND MEASURED PARAMETERS

camera are perfectly aligned. The channel is considered as an analog communications channel where the signal at the receiver is the sampled photocurrents from each image pixel, and do not take into account the quantization limitations in the camera. In practice, measuring photocurrents from a camera image pixel from an integrated camera is non-trivial without breaking open the device.

Camera based communication using LCD screens was studied in [17]. The capacity estimates in this work were based on a water-filing algorithm assuming the camera channel can be equalized to encounter the effects of spatial distortions. But the model and the prototype were designed for a fixed distance of 2m between the screen and camera and did not study the effects of perspective on the estimated capacity and

throughputs achieved.

Perspective distortion has been looked at by the imaging community previously [12], [22], but the fact that the camera is a part of a wireless communication channel presents a new domain of challenge for applying imaging models in analyzing communication channels. In camera based communications, the camera images the source of light (the transmitter), unlike classical computer vision where the camera images an object that reflects light.

The advent of high-resolution cameras in mobile devices has spurred interest in using cameras for communication to retrieve information from screens [1], [14], [18], [20]. These applications use specific receiver processing schemes to combat visual distortions. PixNet [20] proposes to use OFDM modulation to combat the effect of perspective distortion on images by inverse filtering on the estimated channel, and using forward error correction. COBRA [14] proposes to leverage from encoding on the color channels to achieve throughput gains for smartphone screen-camera communication, but at very short distances (22cm). The fact that several prototypes have been constructed reveals that screen-camera communication is gaining large momentum. In this paper, we show that there is still a large room for improvement in throughput of these prototypes in comparison with the capacity of screen-camera communication.

VII. CONCLUSION

In this paper, we discussed the applicability of cameras for communication. We considered the example where cameras could be used as receivers for data transmitted in the form of time-varying 2D barcodes from display screens. We modeled a screen-camera channel using camera projection theory, which addressed visual channel perspective distortions in more detail than prior works. We discussed and modeled the effect of perspective distortion on the information capacity of screen-camera communications. We conducted calibration and validation experiments, and our measurements concurred with the model. Our capacity estimates indicated that, even with the frame-rate limitations in off-the-shelf mobile cameras, data-rates of the order of hundreds of kbps -to- Mbps is

possible even when the 2D barcode from the screen images onto only a small portion of the camera image. While these bounds are much less than the ideal (for example, $8\text{bits/pixel} \times 8\text{Mpixel/frame} \times 30\text{fps}$ for the tablet camera we experimented with), the bound value of data-rates are still promising for medium sized data-transfer or even streaming applications; such as downloading a file from a smartphone screen or streaming a movie from a large display wall. Our estimates indicate that current prototypes have only achieved less than half their capacity, which means that designing efficient techniques to address perspective distortions is still an open problem for building high-data rate camera communications.

VIII. ACKNOWLEDGMENTS

This work is supported by the US National Science Foundation (NSF) under the grant CNS-1065463

REFERENCES

[1] HCCB High Capacity Color Barcodes <http://research.microsoft.com/en-us/projects/hccb/about.aspx>.

[2] Billboard sizes. <http://www.sbuilts.com/sizes.cfm>.

[3] IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs): On Study Group Status for Camera Communications. <http://tinyurl.com/odkrr9w>.

[4] Camera calibration toolbox for matlab. http://www.vision.caltech.edu/bouguetj/calib_doc/index.html#links.

[5] Apple-IPhone 5s-iSight Camera. <http://www.apple.com/iphone-5s/camera/>.

[6] Lecture notes by D.V. Valen: Point Spread Function Workshop <http://tinyurl.com/p88lkr>.

[7] A. Goldsmith. *Wireless Communications*. Cambridge, 2005.

[8] Visible light communication consortium. <http://vlcc.net>.

[9] M. Varga, A. Ashok, W. Yuan, M. Gruteser, N. Mandayam, and K. Dana. Demo: Visual MIMO based LED - Camera Communication Applied to Automobile Safety. In *Proceedings of the 9th international conference on Mobile systems, applications, and services*, 2011.

[10] A. Ashok, M. Gruteser, N. Mandayam, J. Silva, M. Varga, and K. Dana. Challenge: Mobile Optical Networks Through Visual MIMO. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, MobiCom '10, pages 105–112, New York, NY, USA, 2010. ACM.

[11] L. Zheng and D.N.C Tse, Diversity and multiplexing: a fundamental tradeoff in multiple-antenna channels In *IEEE Transactions on Information Theory*, vol. 49, 2003, pages 1073–1096

[12] H. Chen, R. Sukhthankar, G. Wallace, and T. jen Cham. Calibrating scalable multi-projector displays using camera homography trees. In *In Computer Vision and Pattern Recognition*, pages 9–14, 2001.

[13] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Prentice Hall, January 2002.

[14] T. Hao, R. Zhou, and G. Xing. Cobra: color barcode streaming for smartphone systems. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*, MobiSys '12, pages 85–98, New York, NY, USA, 2012. ACM.

[15] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[16] B. K. P. Horn. *Robot vision*. MIT Press, Cambridge, MA, USA, 1986.

[17] S. Hranilovic and F. Kschischang. A pixelated mimo wireless optical communication system. *Selected Topics in Quantum Electronics, IEEE Journal of*, 12(4):859–874, jul. 2006.

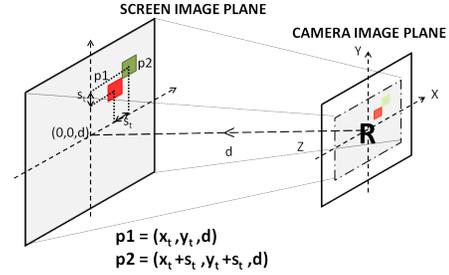


Fig. 8. Illustration Showing the Screen and Camera Image Axis

[18] X. Liu, D. Doermann, and H. Li. A camera-based mobile data channel: capacity and analysis. In *Proceedings of the 16th ACM international conference on Multimedia*, MM '08, pages 359–368, New York, NY, USA, 2008. ACM.

[19] T.S. Lomheim and G.C. Holst. *CMOS/CCD sensors and camera systems*. The International Society for Optical Engineering (SPIE), 2 edition, 2011

[20] S. D. Perli, N. Ahmed, and D. Katabi. Pixnet: interference-free wireless links using lcd-camera pairs. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, MobiCom '10, pages 137–148, New York, NY, USA, 2010. ACM.

[21] A. Tang, J. Kahn, and K.-P. Ho. Wireless infrared communication links using multi-beam transmitters and imaging receivers. In *Communications, 1996. ICC 96, Conference Record, Converging Technologies for Tomorrow's Applications. 1996 IEEE International Conference on*, volume 1, pages 180–186 vol.1, Jun 1996.

[22] R. Yang, D. Gotz, J. Hensley, H. Towles, and M. S. Brown. Pixelflex: A reconfigurable multi-projector display system, 2001.

[23] T. Komine and M. Nakagawa. Fundamental analysis for visible-light communication system using led lights. *IEEE Transactions on Consumer Electronics*, 50(1):100–107, Feb 2004.

APPENDIX

A. Derivation For Perspective Scaling Factor α_p Using Camera Projection Theory

Consider a point $[X_w, Y_w, Z_w]^T$ in world 3D space coordinates with respect to the camera image axis. The 3D coordinates in space map to the corresponding camera image 2D coordinates $[x, y]^T$ through camera projection equations as,

$$\begin{bmatrix} x & y & 1 \end{bmatrix}^T = \mathbf{C} [\mathbf{R} \quad \mathbf{T}] \begin{bmatrix} X_w & Y_w & Z_w \end{bmatrix}^T \quad (11)$$

where T denotes transpose operation, \mathbf{C} , \mathbf{R} , \mathbf{T} are the camera calibration matrix, rotation matrix and translation vector respectively.

Camera calibration matrix \mathbf{C} accounts for the projection and scaling of the coordinates in the image and is a function of the camera focal-length f_{cam} , the side length of each pixel s_{cam} , and the principal point (o_x, o_y) - the coordinates of the camera image center. \mathbf{R} is the rotation matrix that accounts for the 3-tuple rotation angle $(\theta_x, \theta_y, \theta_z)$ of the screen coordinate frame about the X, Y, and Z camera axes, and \mathbf{T} is the translation vector that accounts for the translation between the world coordinate and the camera axis. If $c\theta = \cos \theta$, $s\theta = \sin \theta$ then,

$$\mathbf{R} = \begin{bmatrix} c\theta_z & -s\theta_z & 0 \\ s\theta_z & c\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\theta_y & 0 & s\theta_y \\ 0 & 1 & 0 \\ -s\theta_y & 0 & c\theta_y \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\theta_x & -s\theta_x \\ 0 & s\theta_x & c\theta_x \end{bmatrix} \quad (12)$$

$$\mathbf{C} = \begin{bmatrix} \frac{f}{s_{cam}} & 0 & o_x \\ 0 & \frac{f}{s_{cam}} & o_y \\ 0 & 0 & 1 \end{bmatrix} \quad \mathbf{T} = \begin{bmatrix} T_w^x \\ T_w^y \\ T_w^z \end{bmatrix} \quad (13)$$

Let us consider two adjacent pixels p1 and p2 of the screen transmitter, as shown in Figure 8, whose centers are separated in screen space by one pixel distance of s_t units (side-length of the square pixel). Let x_t, y_t denote the distance of pixel p1 from the screen's center in X and Y dimensions respectively. Let the screen coordinate axis center be situated at a distance d from the camera image axis center. Then using camera projection matrix equation from equation (11), the distortion in each pixel, $\alpha_{(x_t, y_t)}(x, y)$ can be derived as,

$$\begin{bmatrix} x_{p1} \\ y_{p1} \\ 1 \end{bmatrix} = \mathbf{C}[\mathbf{R} \quad \mathbf{T}] \begin{bmatrix} x_t \\ y_t \\ d \end{bmatrix} \quad \begin{bmatrix} x_{p2} \\ y_{p2} \\ 1 \end{bmatrix} = \mathbf{C}[\mathbf{R} \quad \mathbf{T}] \begin{bmatrix} x_t + s_t \\ y_t + s_t \\ d \end{bmatrix} \quad (14)$$

$$\alpha_{(x_t, y_t)}(x, y) = |x_{p2} - x_{p1}| \times |y_{p2} - y_{p1}| \quad \forall (x, y) \in \mathbb{R} \\ = 0, \quad \text{otherwise} \quad (15)$$

$$\alpha_{(x_t, y_t)}(x, y) = s_t \frac{\frac{f_{cam}}{s_{cam}}(c\theta_y + s\theta_x s\theta_y) + o_x(s\theta_y - s\theta_x c\theta_y)}{x_t s\theta_y - y_t s\theta_x c\theta_y + c\theta_x c\theta_y d} \\ \times s_t \frac{\frac{f_{cam}}{s_{cam}}(c\theta_y) + o_y(s\theta_y - s\theta_x c\theta_y)}{x_t s\theta_y - y_t s\theta_x c\theta_y + c\theta_x c\theta_y d} \quad (16)$$

where where $|\cdot|$ denotes the absolute value. Coordinates $x_{p1}, y_{p1}, x_{p2}, y_{p2}$ can take real values but are unit-less quantities. \mathbb{R} denotes the set of camera pixels corresponding to the screen's projected image. The following assumptions were made in our derivation: (a) $s_t \ll d$, the distance between two centers of adjacent screen pixels (order of microns) is negligible when compared to the distance between the camera and screen (few cm to m in typical applications). (b) Rotation about Z axis (see Figure. 8) does not effect pixel distortion. Using equation16 the average distortion factor α_p can be determined as,

$$\alpha_p = \frac{1}{||R_s||} \sum_{\forall (x_t, y_t)} \frac{1}{||R_{cam}||} \sum_{\forall (x, y)} \alpha_{(x_t, y_t)}(x, y) \quad (17)$$

where $||R_s||, ||R_{cam}||$ are the screen and camera resolutions respectively.