NISTIR 5567

An Image-Based Texture-Independent Visual Motion Cue for Autonomous Navigation

Sridhar Reddy Kundur
Robotics Center and
Department of Electrical Engineering
Florida Atlantic University
Boca Raton, FL 33431

Daniel Raviv
Robotics Center and
Department of Electrical Engineering
Florida Atlantic University
Boca Raton, FL 33431
and
Sensory Intelligence Group
Intelligent Systems Division

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards
and Technology
Bldg. 220 Rm. B124
Gaithersburg, MD 20899

NISTIR 5567

An Image-Based Texture-Independent Visual Motion Cue for Autonomous Navigation

Sridhar Reddy Kundur
Robotics Center and
Department of Electrical Engineering
Florida Atlantic University
Boca Raton, FL 33431

Daniel Raviv
Robotics Center and
Department of Electrical Engineering
Florida Atlantic University
Boca Raton, FL 33431
and
Sensory Intelligence Group
Intelligent Systems Division

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards
and Technology
Bldg. 220 Rm. B124
Gaithersburg, MD 20899

January 1995

DEPARTMENT OF COMMERCE
★ UNITED STATES OF AMERICA ★

U.S. DEPARTMENT OF COMMERCE
Ronald H. Brown, Secretary

TECHNOLOGY ADMINISTRATION
Mary L. Good, Under Secretary for Technology

NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
Arati Prabhakar, Director

An Image-Based Texture-Independent Visual Motion Cue for Autonomous Navigation*

Sridhar Reddy Kundur[1] and Daniel Raviv[1,2]
[1]Robotics Center and Department of Electrical Engineering
Florida Atlantic University, Boca Raton, FL 33431
[2]Intelligent Systems Division,
National Institute of Standards and Technology(NIST)
Bldg. 220, Room B124, Gaithersburg, MD 20899
email: kundur@acc.fau.edu and ravivd@acc.fau.edu

Abstract

This paper presents a new visual motion cue, we call the Visual Threat Cue (VTC) that provides some measure for a *relative change in range* as well as *clearance* between a 3D surface and a *fixated* observer in *motion*. We define the VTC, derive a theoretical model for it and show the visual field related to it. This cue is *independent* of the 3D environment, needs almost no *a-priori knowledge* about it, is *rotation independent,* and is measured in [time^{-1}] units.

A practical method to extract the VTC from a sequence of images of a 3D textured surface obtained by a *fixated, fixed-focus monocular* camera in motion is also presented. A *global dissimilarity measure* is extracted *directly from the raw data of the gray level of textured images* from which the VTC is obtained. This approach of extracting the VTC is *independent of the 3D surface texture* and needs no *optical flow information, 3D reconstruction, segmentation, feature tracking* or pre-processing. It needs almost *no camera calibration.* This algorithm to extract the VTC was applied to a set of twelve different texture patterns (of 3D scenes) from the Brodatz's album, where we observed a similar behavior for all the textures.

Key Words: Active Vision, Visual Motion Cue, Visual Navigation

* This work was supported in part by a grant from the National Science Foundation, Division of Information, Robotics and Intelligent Systems, Grant # IRI-9115939

1 Introduction

When dealing with a moving camera-based autonomous navigation system, a huge amount of visual data is captured. For vision-based navigation tasks like obstacle avoidance, maintaining safe clearance, etc., relevant visual information needs to be extracted from this visual data and used in real-time closed-loop perception-action control system. In order to accomplish safe visual navigation several questions needs to be answered, including:

1. What is the *relevant* visual information to be extracted from a sequence of images?
2. How does one extract this information from a sequence of 2D images?
3. How to generate control commands to the vehicle based on the visual information extracted?

This paper is focused on the first two of the above mentioned questions.

Usually the process of driving or flying in a 3D environment involves a human operator. The operator acts in part as a sensory feedback in the perception-action closed-loop control systems to avoid obstacles, maintain clearance, etc. to ensure safe navigation in real time. It becomes a difficult problem to replace the human operator by a vision-based system to achieve similar tasks for the following reasons: In outdoor navigation the environment is usually unknown and unstructured, and the same 3D scene may result in many different images due to changes in illumination conditions, relative distances, orientation of the camera, choice of fixation point, etc., as well as various camera parameters such as zoom, resolution, focus, etc. There is a need for an approach, to obtain relevant visual information about relative proximity in the presence of the above mentioned factors.

The problem of automating vision-based navigation is a challenging one and has drawn the attention of several researchers over the past few years (see for example [1-14]). Usually identifying the surrounding object is not important for such tasks, i.e., is it a tree, mountain or another vehicle; what is more important is whether a particular object is an obstacle or not, i.e., is the camera on a collision course with it, is there enough clearance, etc.? For navigation tasks recovering 3D scene and its attributes may not be necessary as it may contain information which is not relevant for the task at hand. Visual cues such as time-to-contact, looming carry important information about the relative proximity. These cues do not need any reconstruction process which is usually computationally intense and in many cases are sufficient for safe navigation.

It is well established in the literature (computer vision as well as psychology) that optical flow plays an important role in the control of human motion behavior in the environment [15-19]. Several researchers have addressed the use of optical flow as a feedback signal for vision based autonomous navigation (see for example [3,6-10]). An optical flow based theory of how a driver visually controls the braking of an automobile is presented in [12]. In [12] it is also shown that it is possible to control the braking of a vehicle using visual information without measuring the absolute distance, speed or acceleration/deceleration. The looming effect which is the result of retinal expansion of objects has shown to cause defensive reaction in several animals as well as babies [20, 21]. A detailed qualitative as well as quantitative approach to the concept of looming is presented in [10] where several methods to extract looming is also discussed. Application of certain measures of flow field divergence as a qualitative cue for the task of obstacle avoidance is presented in [7]. In [9] the optical flow field is transformed by using a log-polar transformation to extract visual information about time-to-contact [19]. In [3, 8] the variations in peripheral optical flow are employed to guide a mobile robot through obstacles.

The extraction of the local optical flow employs a constraint equation between the local brightness gradients and the two components of the optical flow. Additional constraints are needed to evaluate the complete flow field [22-24]. In addition, the extraction of optical flow from a sequence of images needs pre-processing like spatio-temporal smoothing which may be computationally expensive. Alternatives to optical flow information as sensory feedback for obstacle avoidance include geometrical properties like size, shape, contour and area of image entities, imaged texture, focus, etc.

In [25] a frequency based texture operator is employed to classify the characteristics of the Fourier transforms of local image windows, to compute the gradients of texture in the image in order to get depth information. A differential invariant of the image field based visual information about time-to-collision is presented in [14]. Variations in image statistics are employed to extract the four components of an affine transformation are presented in [26]. A qualitative view of the use of these components as sensory feedback information for collision avoidance is also presented [26]. In [27] it is shown that the relative changes in edges of visible texture in a unit area to be equal to looming described in [10]. This approach of using edge density in an image is an alternative to the use of flow based approach to extract looming which is very sensitive to noise.

This paper presents a new visual motion cue, we call the Visual Threat Cue (VTC) that provides some measure for *relative change in range* as well as *clearance* between a 3D surface and a *fixated* observer in *motion*. In this paper we define the VTC, derive a theoretical model for it and present a robust practical way to extract it from a sequence of images. This cue is *independent* of the 3D environment and needs no *a-priori knowledge* about it. It is *rotation independent,* and is measured in [time^{-1}] units. It is also scale independent but we do not show it here.

The VTC corresponds to a visual field surrounding a moving observer, i.e., there are imaginary 3D surfaces attached to the observer that are moving with it, each of which corresponds to a value of the VTC. The points that lie on a relatively smaller imaginary *equal* VTC *surface* correspond to a relatively larger value of the VTC, indicating a relatively higher threat of collision. Value of the VTC is infinity when the radial clearance between the surface and the observer is equal to the minimum clearance desired, which gives an indication of *absolute range* between the observer and the surface. A positive value of the VTC corresponds to the region in front of the observer and a negative value corresponds to the region in back of the observer. By measuring the VTC of surfaces a measure of threat is obtained from which a control command can be generated.

A practical method to extract the VTC from a sequence of images of a 3D textured surface obtained by a *fixated, fixed-focus monocular* camera in motion is also presented. For each image in such a 2D image sequence of a textured surface, a *global variable* we call the Image Quality Measure (IQM) is obtained *directly from the raw data of the gray level images*. The VTC is obtained by calculating relative temporal changes in the IQM. This approach of extracting the VTC is *independent of the 3D surface texture* and needs no *optical flow information*, *3D reconstruction*, segmentation, *feature tracking*, pre-processing and can be extracted directly from the raw gray level data of images. The process of extraction can be seen as a *sensory fusion* of *focus*, *texture* and *motion* at the *raw data level and* needs almost *no camera calibration*. This algorithm works better on images obtained from natural scenes including fractal-like images, where more details of the 3D scene are visible in the images as the range shrinks and also can be implemented in parallel hardware. This algorithm to extract the VTC was applied to a set of 12 different textures from the Brodatz's album, where we observed that the relative temporal variations in the IQM behaves in a similar fashion and its value is approximately equal for most of the textures employed. A comparison of the theoretical VTC and the VTC extracted from the images is presented.

This paper is organized as follows: In section 2 we define the VTC and its relation to several other visual cues like time-to-contact, looming and also derive a theoretical model for it. In section 3 we describe the procedure to extract the IQM and the VTC from a sequence of images and in section 4 experimental results and the analysis are presented.

2 The Visual Threat Cue (VTC)

In this section we define the VTC, present the equal VTC surfaces, explain its significance as a sensory feedback signal and also derive a relation between the VTC and the radius of the blur circle

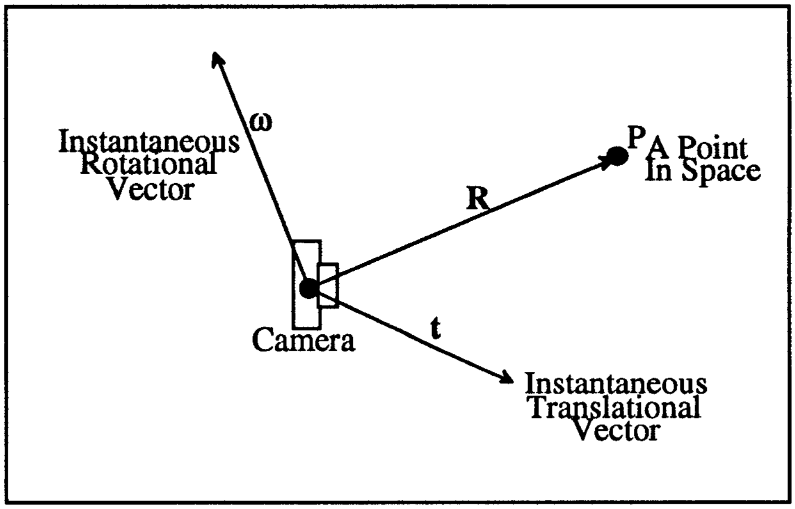
2.1 Definition

Mathematically we define the VTC, for R > R_0 as follows

$$VTC = R_0 \frac{\frac{d}{dt}(R)}{R(R - R_0)} \qquad (1)$$

Where R is the range between the observer and a point on the 3D surface, d(R)/dt is the differentiation of R with respect to time and R_0 is the desired minimum clearance and has the same units as R. We explain the significance of R_0 in later sections. Note that the units of the VTC are [time^{-1}].

2.2 The VTC in Vector Form



ω
Instantaneous Rotational Vector
P
A Point In Space
R
Camera
t
Instantaneous Translational Vector

Figure 1: Motion Parameters

Assume a moving observer in a stationary environment. Let the instantaneous vector from the observer to a stationary point P be R, the instantaneous translational velocity vector of the camera be **t** and the angular velocity vector of the camera be ω in camera coordinates as shown in Figure(1).

Since

$$\frac{d}{dt}(R) = \frac{\boldsymbol{V} \bullet \boldsymbol{R}}{|\boldsymbol{R}|^2}$$

where V is the velocity vector of the point P in the camera coordinate system, i.e.,

$$\boldsymbol{V} = -\boldsymbol{t} - \omega \times \boldsymbol{R}$$

Equation (1) can be written as follows:

$$VTC = R_0 \frac{(-\boldsymbol{t} - \omega \times \boldsymbol{R}) \bullet \boldsymbol{R}}{|R|^2 |R - R_0|} \tag{2}$$

since $(\omega \times \boldsymbol{R}) \bullet \boldsymbol{R} = 0$, Equation (2) reduces to

$$VTC = R_0 \frac{(-\boldsymbol{t}) \bullet \boldsymbol{R}}{|R|^2 |R - R_0|} \tag{3}$$

Note that the VTC in Equation (3) is a scalar and is dependent only on the observer's translational velocity, but is *independent* of its rotational component. In a later section we describe a practical way to extract this VTC from a sequence of monocular images.

The same holds when both the camera and the 3D point are in motion since any relative motion can be represented as motion of the camera relative to a stationary environment.

2.3 Equal VTC surfaces

In this section we provide simulation results to show the location of points beyond the desired minimum clearance R_0 in 3D space around an observer in motion, that have the same value of the VTC for any motion of the camera. The VTC corresponds to a visual field surrounding the moving observer, i.e., there are imaginary 3D surfaces attached to the observer that are moving with it, each of which corresponds to a value of the VTC.

The points that lie on a relatively smaller surface corresponds to a relatively larger value of VTC, indicating a relatively higher threat of collision. The VTC on the minimum clearance sphere of R_0 centered at the location of the observer is the maximum which is infinity, indicating that the absolute distance between the observer and the camera is the minimum clearance. A positive value of VTC corresponds to the region in front of the observer and a negative value corresponds to the region in back of the observer. A section of this visual field is shown in Figures (2). Note that this field is not a a sphere in 3D. However it is symmetric about the instantaneous translational vector **t**.

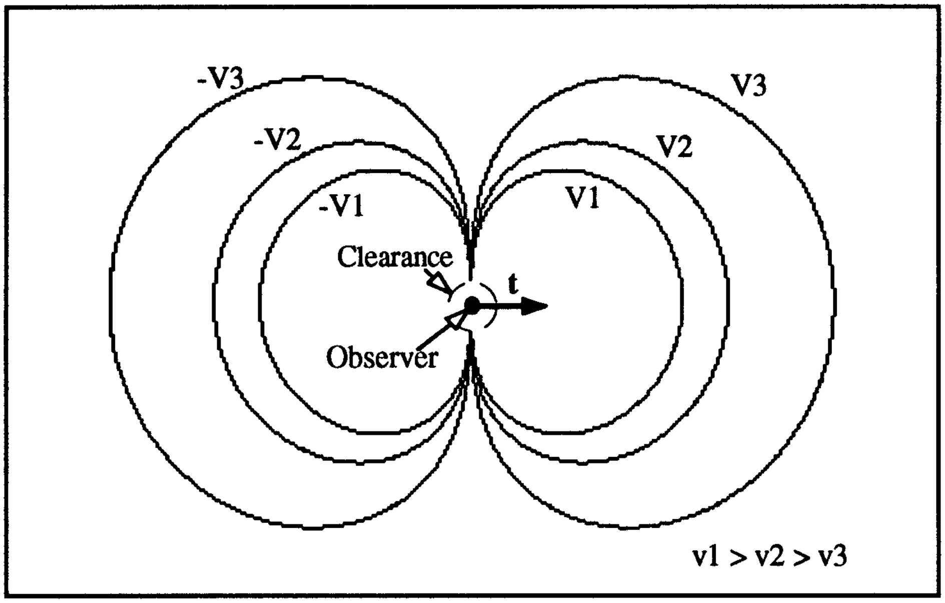


Figure (2): Cross section of the VTC, **t** is the instantaneous translational vector

2.4 The VTC and 3D Space

As mentioned in sections 2.1 and 2.2, the size of equal VTC surfaces is a function of time (i.e., measured as [time^{-1}] units) and are not measured in distance units. This means that different surfaces in 3D may result in the same VTC value for different translational velocities. For a given direction, a point located at a greater distance with a relatively larger value of the relative translational velocity vector may have the same value of the

VTC as a point at a closer distance with relatively smaller value of translational velocity vector.

2.5 The VTC as a Sensory Feedback Signal

The VTC divides the 3D space around the observer in motion into 3D regions. Points in front of the observer correspond to positive values of the VTC and points in the back of the observer correspond to negative values of the cue. Points that lie on a smaller VTC surface correspond to a larger value in the cue and vice versa. Based on this knowledge about the cue, one can demarcate the region around an observer into safe and danger zones (Figure 3a) or safe, high risk, and safety zones (Figure 3b). A positive value indicates a decrease in the relative range, and if this value exceeds the threshold for a safe zone then appropriate control action has to be taken based on the actual value to accomplish the tasks such as collision avoidance, maintenance of clearance, etc.

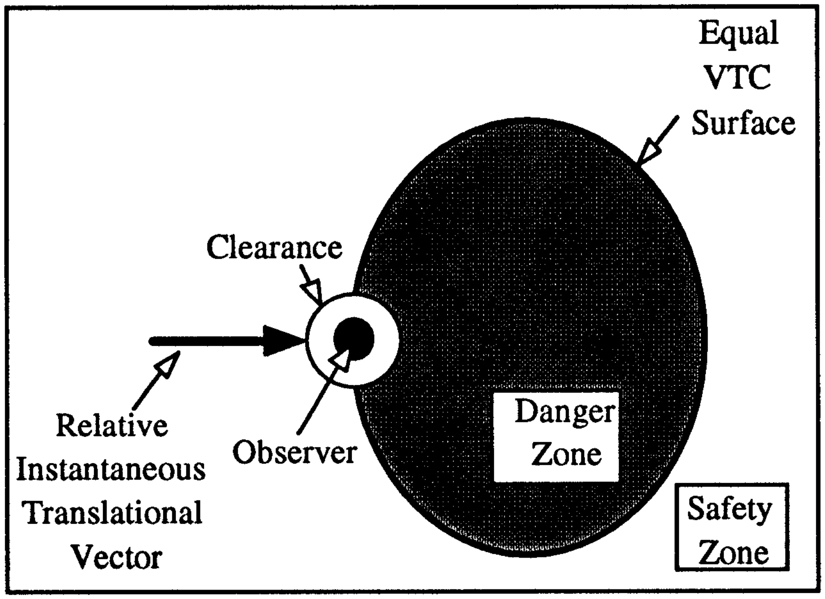


Figure 3(a): Qualitative VTC

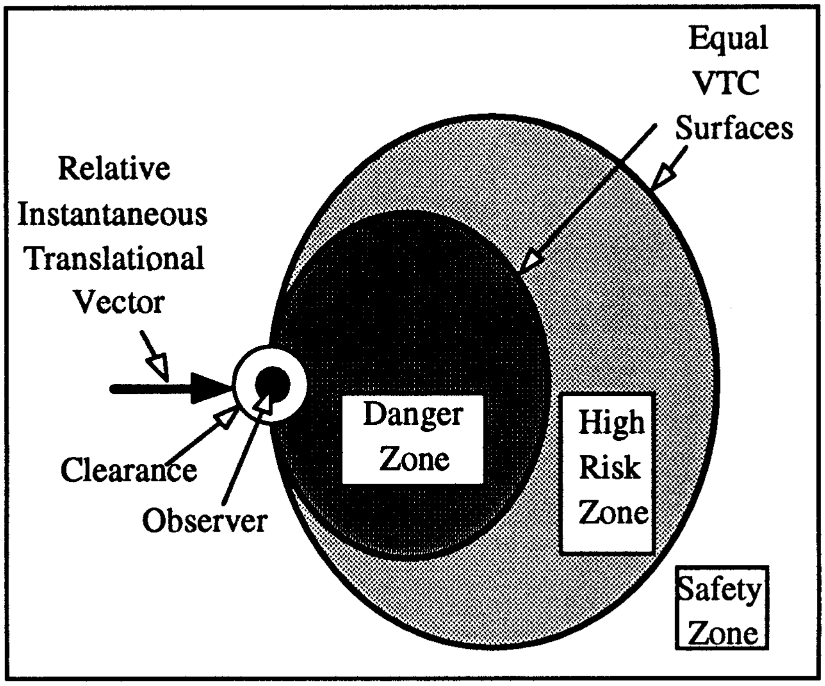


Figure 3(b): Qualitative VTC

2.6 The VTC and The Time-to-Contact

The VTC is related to the concept of time-to-contact presented in [12]. According to [12] any point that lies on a plane which is perpendicular to the instantaneous translational motion direction of the camera, will produce the same value of "time-to-contact" Tc (assuming that the optical axis coincides with the direction of motion). This means that the time-to-contact deals with *depth*. The derivation in [12] is valid only for rectilinear motion (with no rotation) of the camera.

One problem with the time-to-contact approach is that points which lie on a single perpendicular plane but located far away from, or close to, the camera produce the same values of time-to-contact even though they are not equally relevant to making vision-based behavioral decisions. In addition, it is rotation dependent. An advantage of the VTC derived in the previous section is that it is related to *range* rather than *depth,* i.e., deals with more relevant information. As mentioned earlier the VTC is *rotation independent* and deals with *clearance* as well.

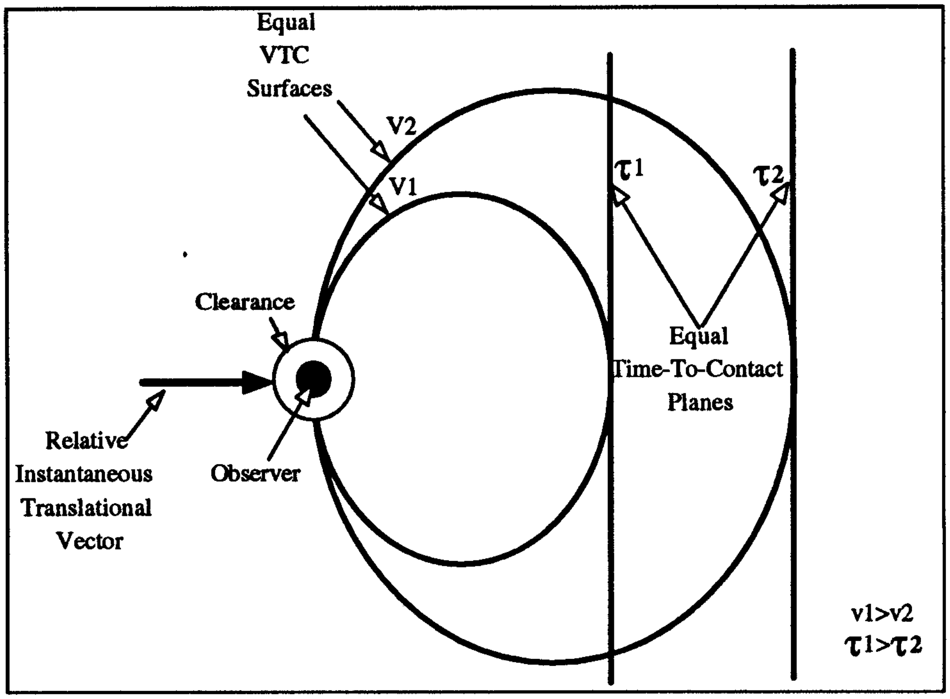


Figure (4a): The VTC and Time-to-Contact

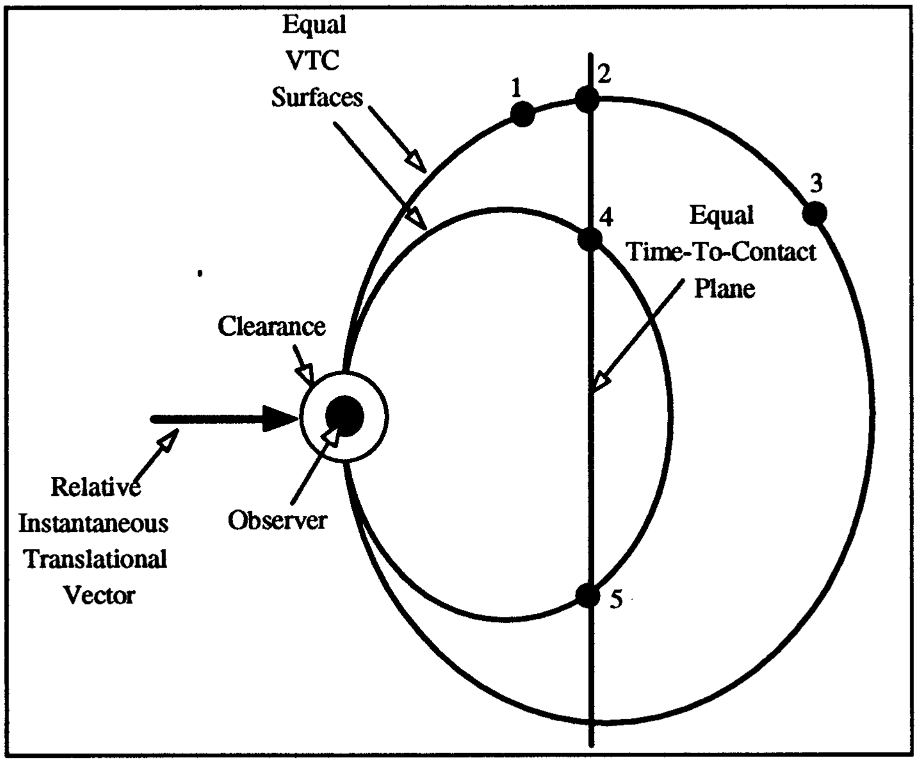


Figure (4b): The VTC and Time-to-Contact

Figures (5a) and (5b) illustrate the main difference between the time-to-contact value of a point, and its VTC. All points that lie on a vertical plane will have the same time-to-contact value. However the points that lie on an equal VTC surface produce the same VTC value. In Figure (5b) points 1, 2 and 3 have the same VTC value but different time-to-contact values. Points 2 and 4 have the same time-to-contact value but different VTC values. Points 4 and 5 have the same time-to-contact values and the same VTC values.

2.7 The VTC and The Looming

The VTC described in this paper is related to the concept of visual looming presented in [10]. The visual looming effect is the result of the expansion of object's size in the retina of the observer. In [10] a detailed qualitative and quantitative analysis of looming is presented. The visual fields associated with looming are shown in [10] to be imaginary *spheres* attached to the observer, whose centers lie on the relative instantaneous translational velocity vector and are moving with the observer (see Figure (5)). One difference between the VTC field and looming field is that the imaginary surfaces in the case of looming are spheres and in the case of the VTC they are not spheres. The equal looming spheres and the equal VTC surfaces are symmetrical about the instantaneous translational velocity vector. In addition the VTC is *defined beyond a certain minimum desired clearance* from the observer and *practically* allows to maintain a desired clearance without measuring it. The definition of looming is valid at any point in space and can not maintain a desired absolute clearance. Both the visual looming and the VTC are dependent on the relative translational motion and are independent of the rotation. Both these cues can be used to increase the reliability of visual sensing for vision based autonomous navigation.

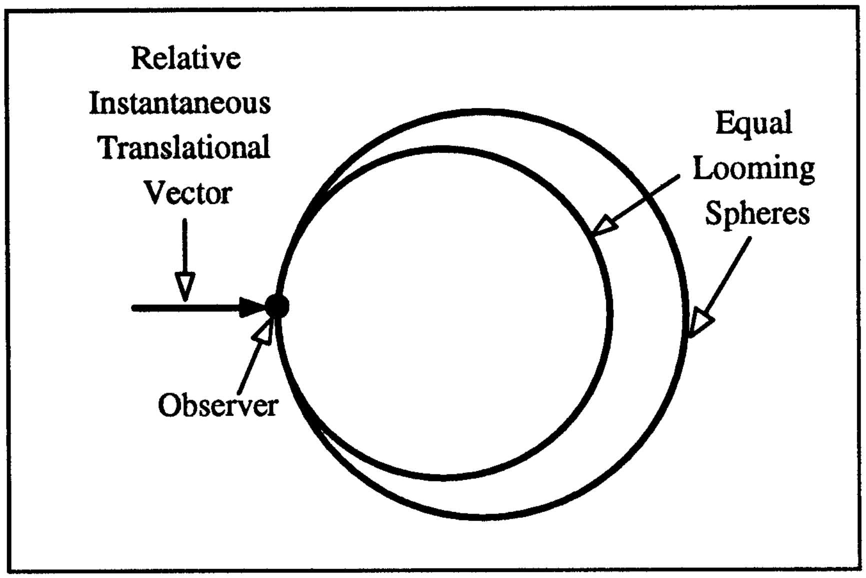


Figure 5 : Qualitative Looming

2.8 Relation between the VTC and the radius of the blur circle

For a given camera, the focal number f, focal length F are constants and for a fixed focus setting the distance between the lens and the image plane v_0 is also constant (see Appendix A). With these constants the following relation can be derived from Equation (A4) from appendix A:

$$\frac{\frac{d}{dt}(\sigma)}{\sigma} = u_0 \frac{\frac{d}{dt}(R)}{R(R-u_0)} \tag{4}$$

where σ is the radius of the blur circle, R is the range between the camera and the fixation point on the object, u_0 is the distance to which the camera is focused initially and the d(.)/dt represents differentiation with respect to time.

The entity on the right hand side of Equation(4) is similar to the VTC described in the previous section and has the units of [time^{-1}]. If R_0 is set to be u_0 which is the distance to which the camera is focused initially, then it becomes the VTC i.e., the desired minimum clearance will be the distance to which the camera is initially focused to. Based on Equation (4) in order to measure $(\frac{\frac{d}{dt}(R)}{R(R-u_0)}u_0)$ from a sequence of images, we need to measure σ and d(σ)/dt. This has been shown to be practically difficult [28-30]. In later sections we present an alternate way to measure $(\frac{\frac{d}{dt}(R)}{R(R-u_0)}u_0)$ in a robust manner with out measuring σ and d(σ)/dt.

3 Extraction of the VTC from a sequence of Images

In this section we present a practical way to extract the VTC described in Equations (1) and (4) from a sequence of images, using temporal variations of the *IQM*. It is extracted directly from the raw gray level data without measuring σ and d(σ)/dt.

3.1 Image Quality Measure (IQM)

Local spatial gray tone variations in an image give rise to a visual pattern in the image known as texture. These spatial gray level variations are due to the visual characteristics of the 3D scene being imaged, the illumination, the range between the scene and the observer, as well as due to camera parameters like zoom, aperture, resolution, focus, etc. When there is a relative motion between a textured surface and a fixated, fixed-focus moving observer, the perceived texture in the 2D image varies. For instance, consider the case of a camera that is initially focused to a 3D surface at a very short distance and gradually moves away from this surface. As a result, the perceived 2D image texture varies from one image to another, mainly due to focus, i.e., the image of the scene in perfect focus is very sharp and has many details, then as the camera moves away from the scene, fine details gradually get smeared and eventually disappear (see Figure 8). When the image is in perfect focus, the dissimilarity, i.e., spatial gray level variations is very high, and as the details get smeared the dissimilarity gets smaller and smaller. We describe an IQM to measure the dissimilarity of the image. Using the relative temporal variations in this IQM we extract the VTC. Next we present a brief overview of several possible approaches to extract the dissimilarity of images, and describe a practical way to extract the VTC from variations in the IQM.

The area of texture classification has drawn the attention of researchers in the area of computer vision over the past two decades (see for example [31-39]). One of the earliest areas of interest in the texture analysis was texture segmentation and scene analysis. These approaches may be broadly classified as statistical approaches and approaches based on structural properties. The statistical approaches usually employ features that measure the coarseness and the directionality of textures in terms of the averages over a windowed portion of the image. While structural methods on the other hand describe the geometrical properties like size, shape, area, etc. of the objects in the scene. Structural analysis is suitable for structured environments and also needs some a-priori knowledge about the scene. The statistical methods that are in use are mainly classified into the following categories:

1. Spatial gray level dependence methods [31, 33, 34]
2. Spatial frequency based methods [33, 35]
3. Stochastic model based features [36, 37]
4. Heuristic approaches [38, 39]

The most popular being the spatial gray-level dependence and the stochastic model-based approaches. Usually spatial gray level based approaches are probabilistic, dependent on the number of gray levels in the image and computationally expensive. Stochastic model approaches are dependent on the model of the texture. Among several possible approaches to describe the quality of texture in an image, we extended an approach presented in [40] to describe the dissimilarity of images using IQM. The advantages of using this approach over the other approaches are:

1. It gives a *global* measure of quality of the image, i.e., *one* number which characterizes the image dissimilarity is obtained.

2. It does not need any preprocessing, i.e., it works directly on the raw gray level data without any spatial or temporal smoothing.

3. It does not need a model of the texture and is suitable for many textures.

4. It is simple and can be implemented in real time on parallel hardware.

5 It is non-probabilistic and is independent of the number of gray levels used in the image

Mathematically, the IQM is defined as follows (see Appendix B):

$$IQM = \frac{1}{|D|} \sum_{x=x_i}^{x_f} \sum_{y=y_i}^{y_f} \left(\sum_{p=-L_c}^{L_c} \sum_{q=-L_r}^{L_r} \left| I(x,y) - I(x+p, y+q) \right| \right) \tag{5}$$

where I(x,y) is the intensity at pixel (x,y) and x_i and x_f are the initial and final x-coordinates of the window respectively ; y_i and y_f are the initial and final y-coordinates of the window in the image respectively and Lc and Lr are positive integer constants; and D is a number defined as $D=(2L_c+1)\times(2L_r+1)\times(x_f-x_i)\times(y_f-y_i)$. One can see from Equation (5) the IQM is a measure for the dissimilarity of gray level intensity in the image. In our experiments we arbitrarily chose a window of size 50 x 50 pixels in the center of the image and Lc = 5 and Lr = 4.

3.2 Extraction of the VTC from relative variations of the IQM

The IQM described in Equation (5) was applied to a set of 12 different textures from Brodatz's album [42]. The experimental details are provided in the following section. Based on our experimental results, we observed that the IQM mentioned above is almost a constant when the range between the surface and the camera is very large and it increases non linearly as the camera approached the distance to which it is focused to. We observed that the radius of the blur circle varies inversely with the IQM, i.e., when the texture details are sharp, IQM is very high and the radius of blur circle is almost zero, and vice-versa. Hence for ranges greater than the initial distance to which the camera is focused to, we modeled the radius of the blur circle in terms of IQM as follows:

$$IQM \propto \frac{1}{\sigma} \tag{6}$$

or:

$$IQM = \frac{\sigma_0}{\sigma} \tag{7}$$

where σ_0 is some proportionality constant and σ is the radius of the blur circle.

From Equation (7):

$$-\frac{\frac{d}{dt}(\sigma)}{\sigma} = \frac{\frac{d}{dt}(IQM)}{IQM} \tag{8}$$

By combining Equation (4) with Equation (8), using $u_0 = R_0$, we obtain the following relation

$$-\frac{\frac{d}{dt}(IQM)}{IQM} = R_0 \frac{\frac{d}{dt}(R)}{R(R - R_0)} \tag{9}$$

The VTC obtained by using Equation (9) does not need knowledge about the camera parameters like the focal number f or the focal length F and is independent of the magnitude of the IQM.

4 Experimental Details

Several experiments were performed to study the variations in the IQM of image sequences in order to extract the VTC. The system used in the experiments include a Coordinate Measuring Machine (CMM), a CCD video camera, a 486 based personal computer, ITEX PC-VISION PLUS image processing system and several texture plates from Brodatz's album [42]. A block diagram of the connections is shown in Figure (6a).

4.1 Procedure

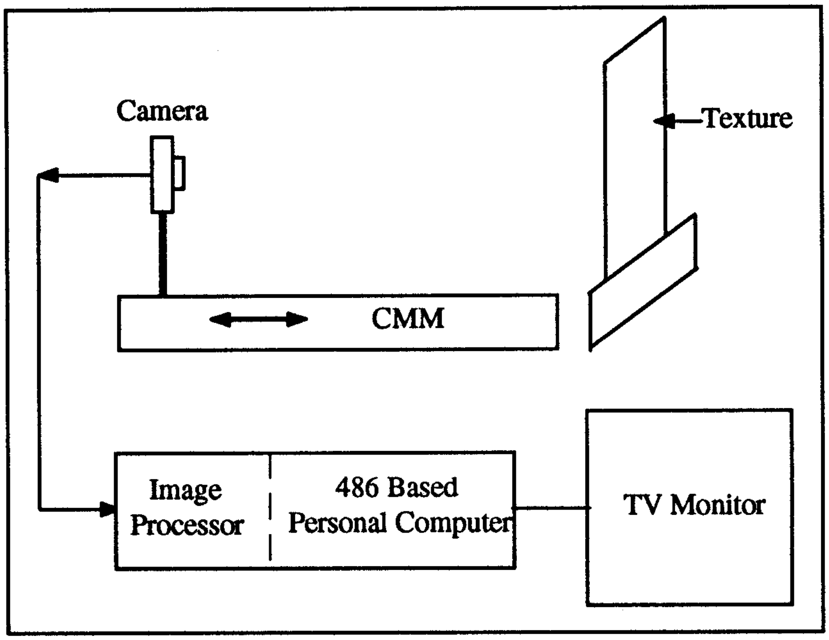


Figure (6a): Block diagram of the Experimental setup

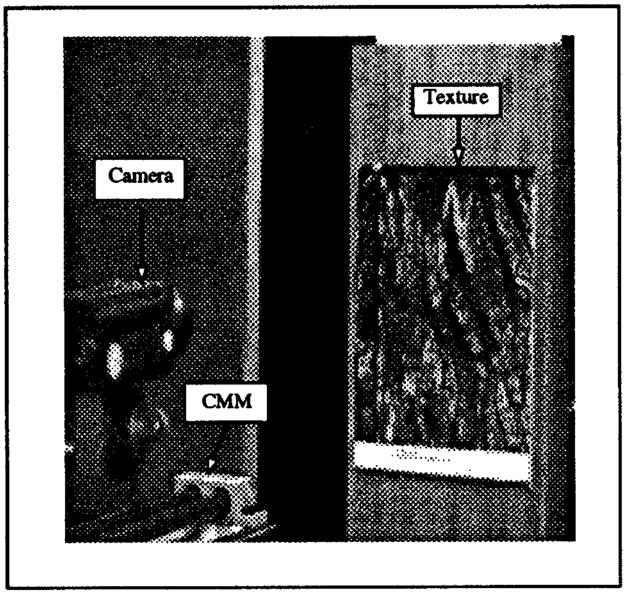
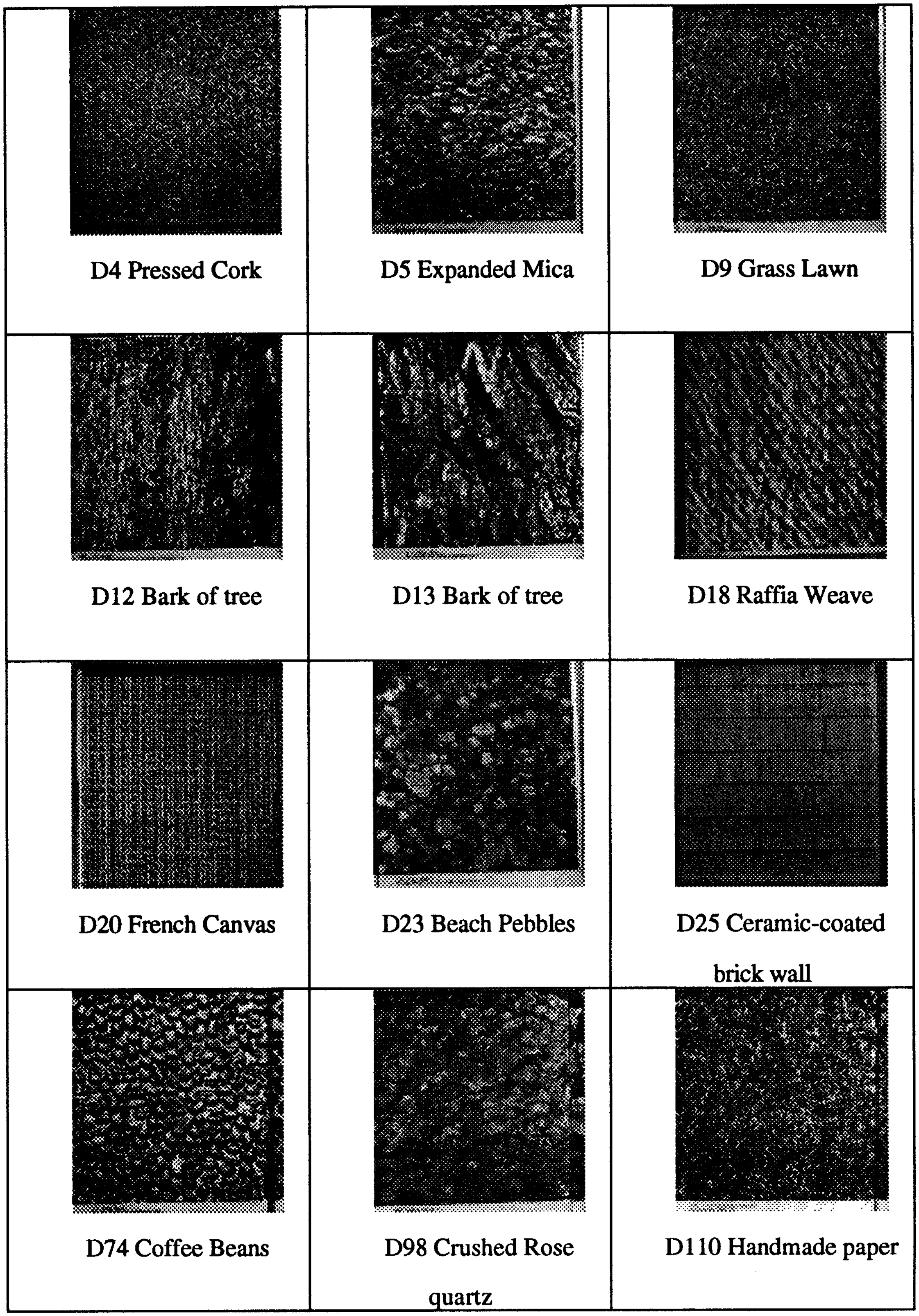


Figure (6b): CMM Set up

A CCD camera is attached to the CMM and the texture surface is placed in front of the camera as shown in the Figure (6b). The maximum distance between the surface and the camera is 900 mm and the minimum distance being 200 mm. The camera is focused to the closest possible distance which in the case of the camera used is 200 mm, i.e., texture details are sharp when the distance between the camera and the surface is 200 mm. The error in the initial setting is about 1 mm. Once this is set, the measurements in relative ranges (for obtaining the ground truth values) were as accurate as the CMM. With this focus setting, the distance between the camera and the surface is varied from 900 mm to 200 mm in steps of 10 mm.

The CCD camera attached to the CMM as shown in Figure (6a) and Figure (6b) captures the images of the texture. These images are then digitized by the PC-based image processor PC-VISION PLUS. These digitized images are then processed by a 486-based personal computer, to extract the IQM and the VTC. For a given texture, we computed these measures at 71 different distances and this was repeated for 12 different textures (shown in Figure (7)) from Brodatz's album [42]. Figure (8) shows an imaged texture (D18 from [42]) as a function of range for 20 different ranges. This set of images shows intuitively the evolution of details in the image as a function of range for a fixed focus camera. The experimental results along with the textures used are presented in the following section.

We have also performed a few experiments to study the variations in the IQM for various initial focus settings. We present the cases for which the initial focus setting was 250 mm, 300 mm and 350 mm. When the initial focus is set to 250 mm, the maximum distance between the camera and the surface is 1060 mm and the minimum distance is 250 mm. When the initial focus setting is 300 mm, the maximum distance between the camera and the surface is 1110 mm and the minimum being 300 mm and in the case where the initial setting is 350 mm the maximum distance between the camera and the surface is 1160 mm and the minimum distance is 350 mm.



D4 Pressed Cork

D5 Expanded Mica

D9 Grass Lawn

D12 Bark of tree

D13 Bark of tree

D18 Raffia Weave

D20 French Canvas

D23 Beach Pebbles

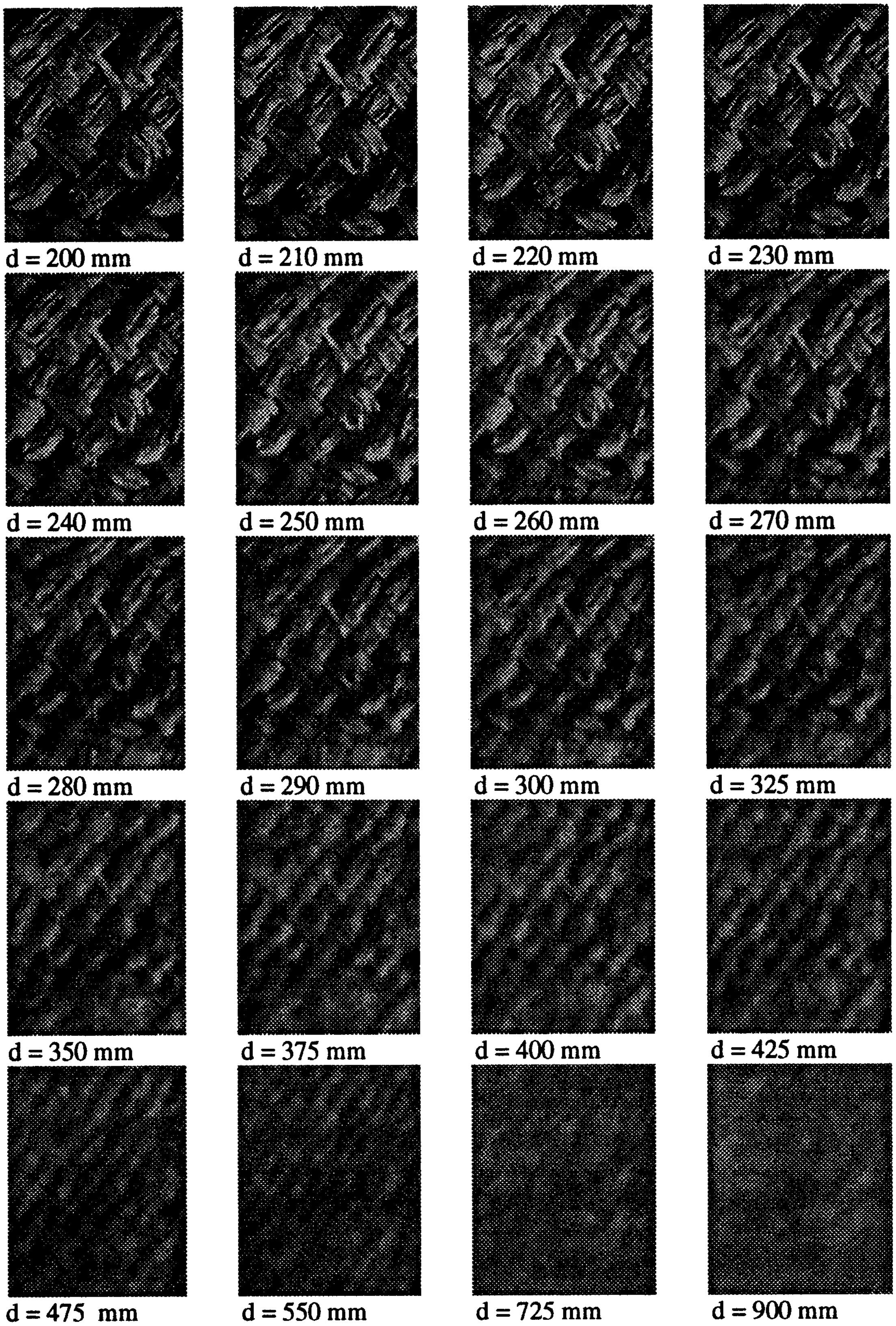
D25 Ceramic-coated brick wall

D74 Coffee Beans

D98 Crushed Rose quartz

D110 Handmade paper

Figure (7): Various texture patterns used in the experiments



d = 200 mm d = 210 mm d = 220 mm d = 230 mm

d = 240 mm d = 250 mm d = 260 mm d = 270 mm

d = 280 mm d = 290 mm d = 300 mm d = 325 mm

d = 350 mm d = 375 mm d = 400 mm d = 425 mm

d = 475 mm d = 550 mm d = 725 mm d = 900 mm

Figure (8): Sequence of images depicting the evolution of details in the image a decrease in the relative range, d is the range between surface (D18) and the camera

4.2 Results and Analysis

The IQM described in Equation (7) is extracted according to the procedure described in the section 3.1, and the VTC is extracted from the relative temporal variations of the IQM.

For each of the texture patterns employed, we present the following:

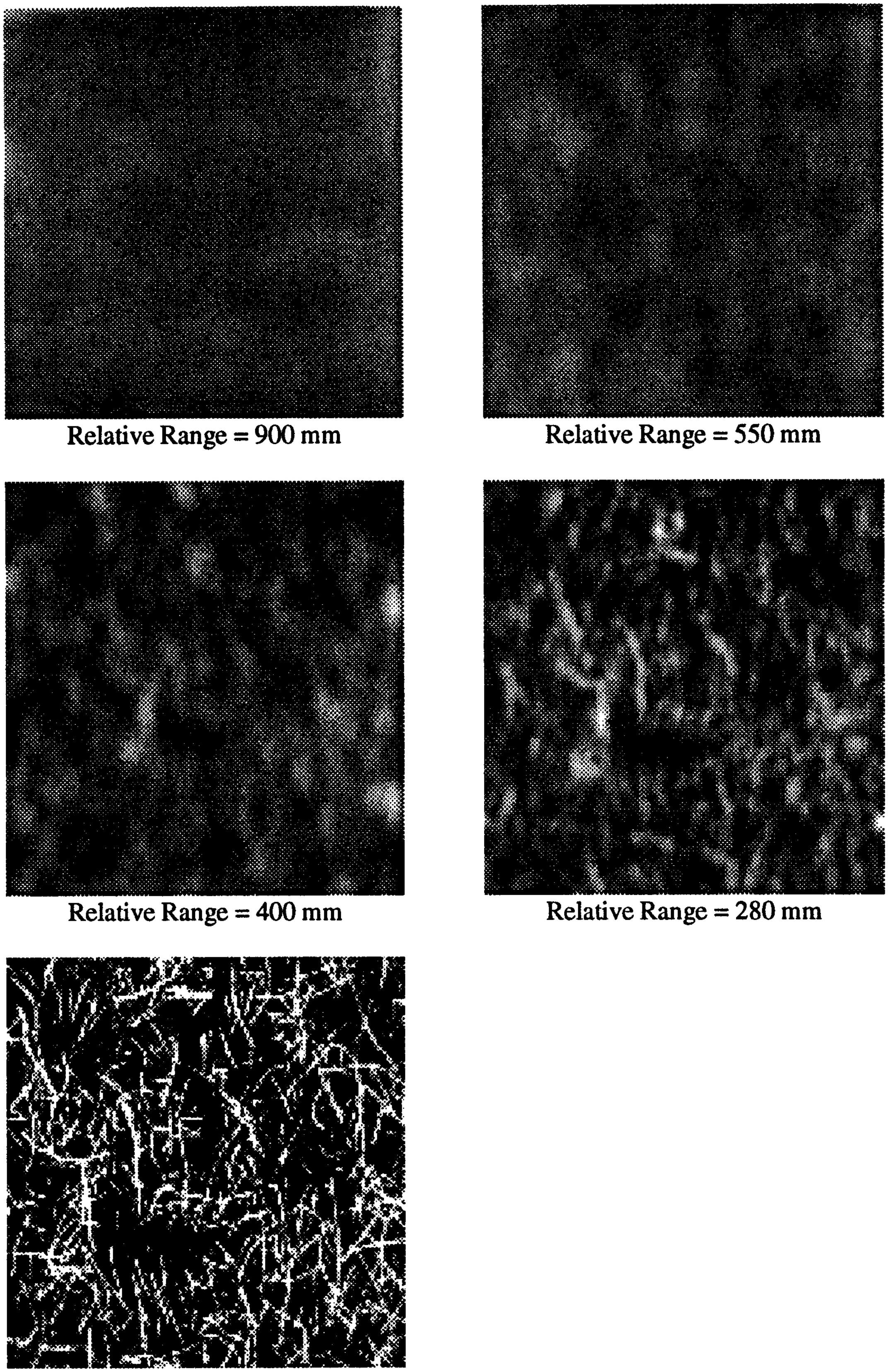
1. Five sample images (out of a total 71 images) relative ranges 200 mm, 280 mm, 400 mm, 550 mm, 900 mm (Figures 9(a)-20(a)).

2. The normalized measured IQM as function of the distance between the camera and the surface (It is normalized since the extraction of the VTC is independent of the absolute magnitude of the IQM. Figures 9b-20b).

3. A plot depicting the theoretical VTC and the VTC extracted from the images (Figures 9c-20c).

Note that $\Delta t = 1$ s in the computation of the VTC, and there is no spatial smoothing on the images or temporal smoothing in the IQM or the VTC.

In Figure (21) , (22) and (23) we show the effect of the desired minimum clearance R_0 on the relation between the IQM and the VTC as a function of distance between the camera and the surface. Note that behavior as a function of the relative range is similar for all clearance values



Relative Range = 900 mm

Relative Range = 550 mm

Relative Range = 400 mm

Relative Range = 280 mm

Relative Range = 200 mm

Figure (9a): Image sequence for texture D110

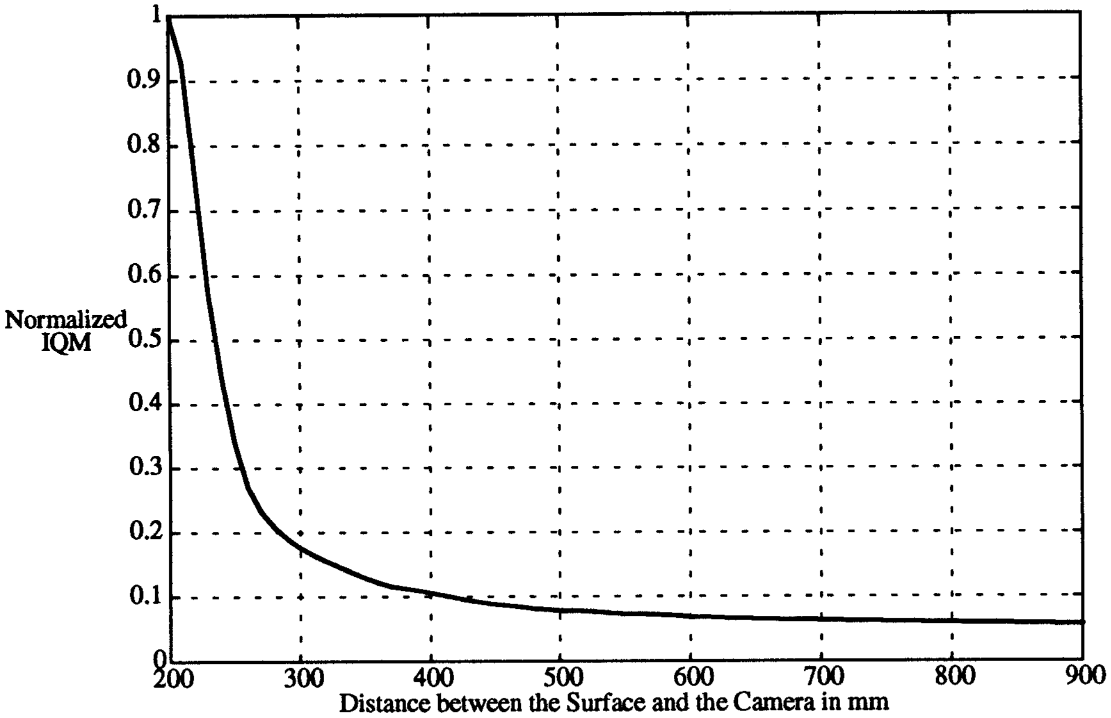


Figure (9b): Normalized IQM versus the distance between the Surface and the Camera for texture D110

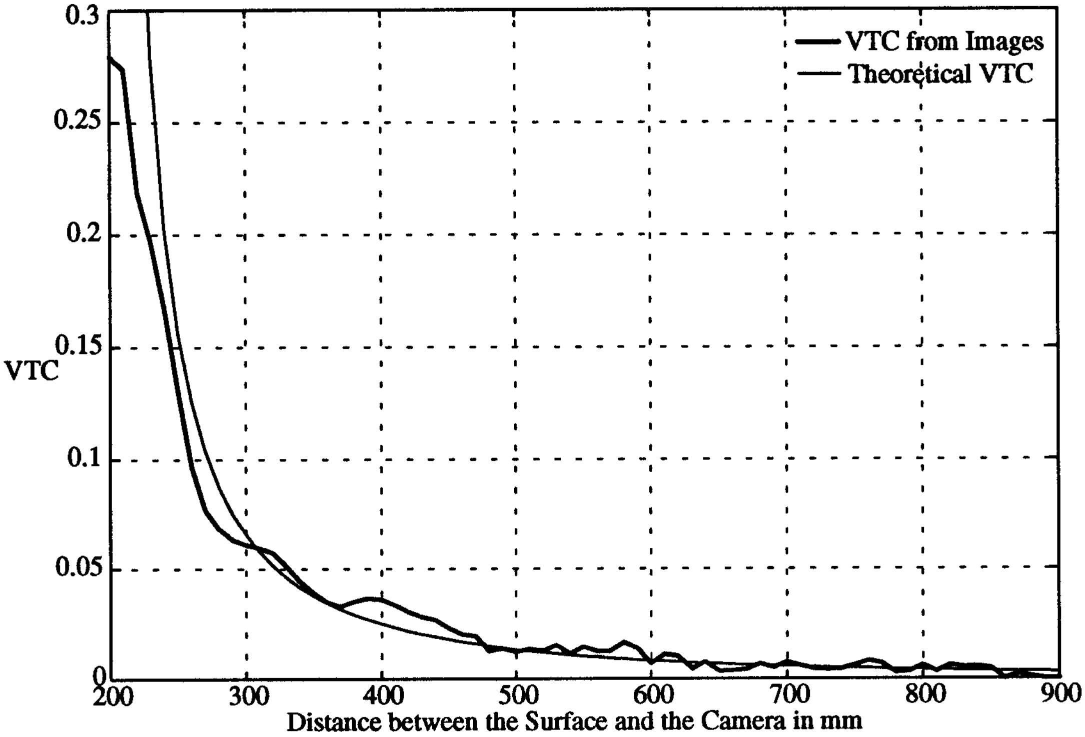
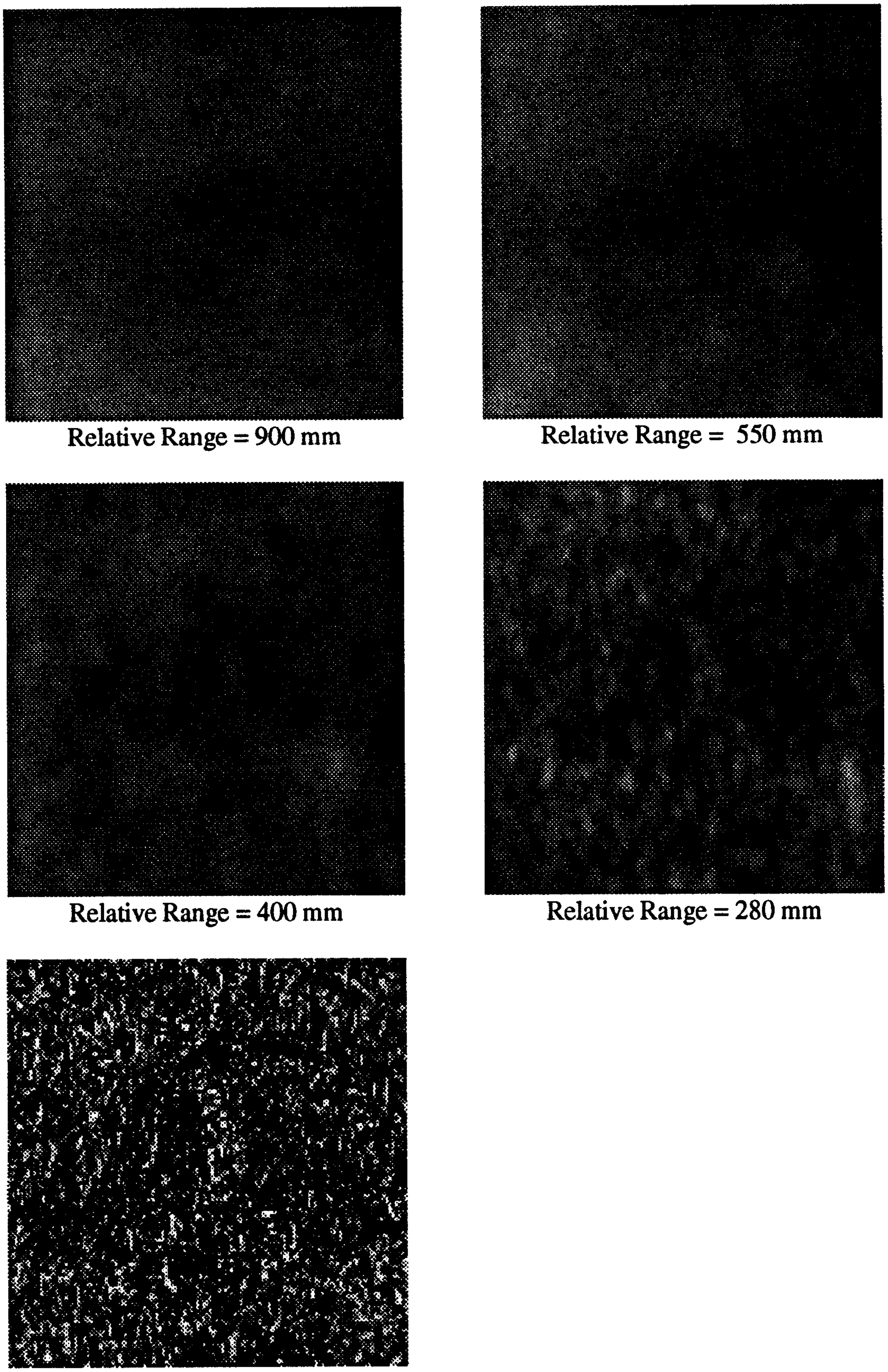


Figure (9c): The VTC versus the distance between the Surface and the Camera for texture D110



Relative Range = 900 mm

Relative Range = 550 mm

Relative Range = 400 mm

Relative Range = 280 mm

Relative Range = 200 mm

Figure (10a): Image sequence for Texture D9

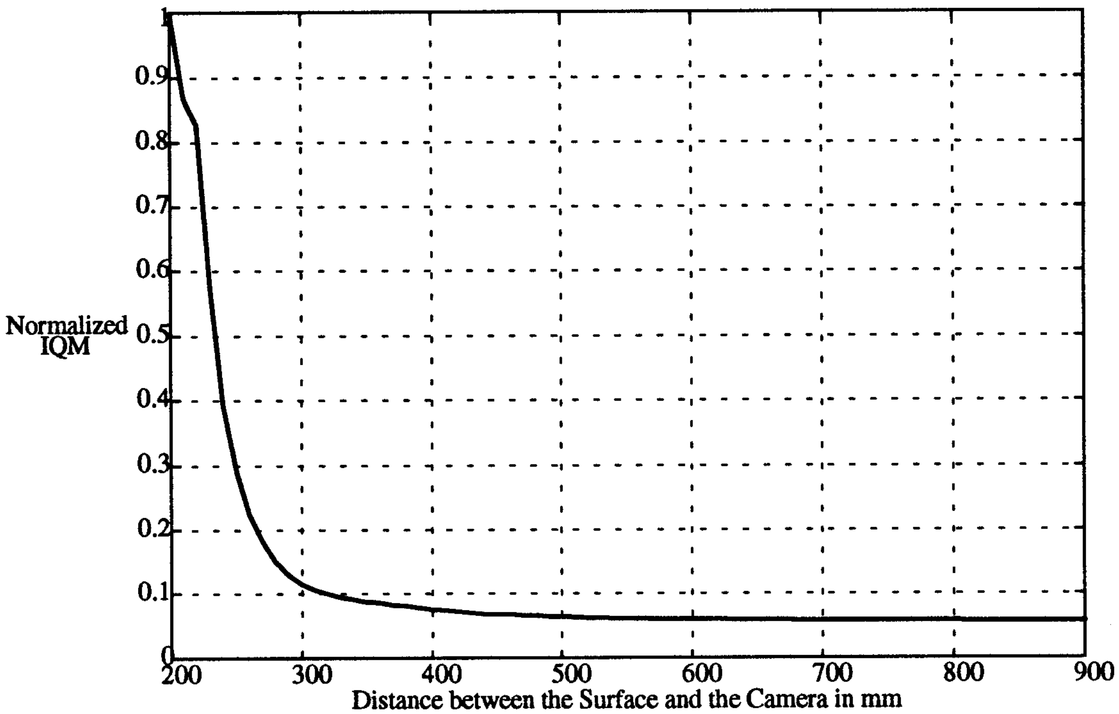


Figure (10b): Normalized IQM versus the distance between the Surface and the Camera for texture D9

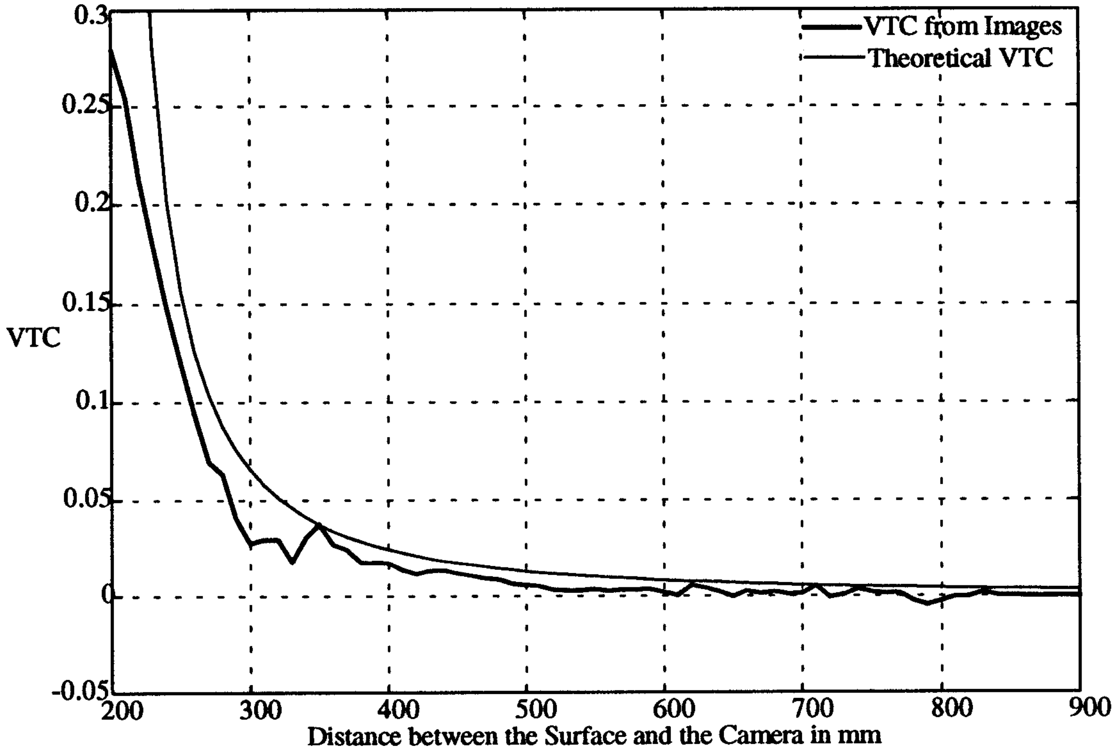
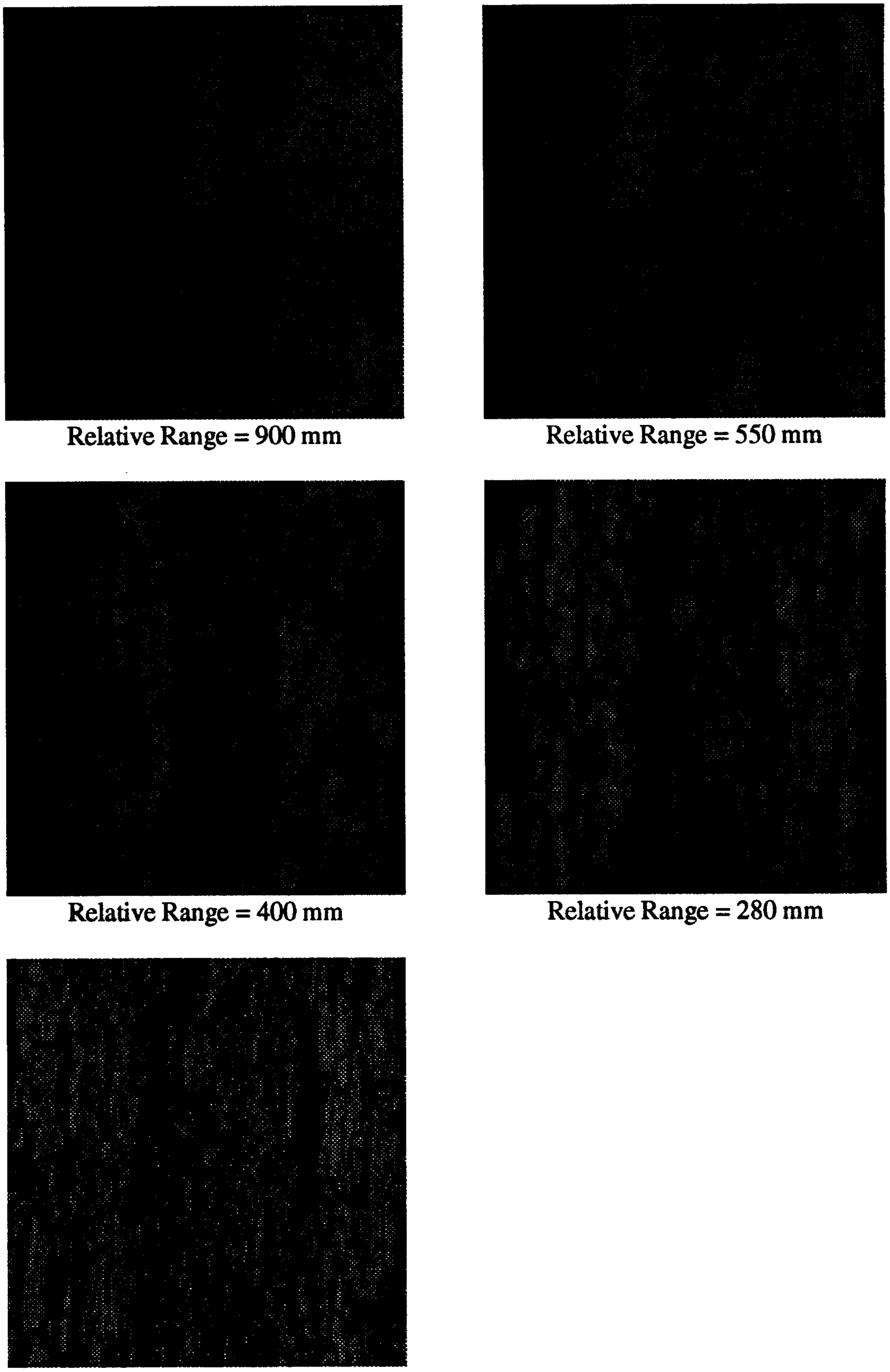


Figure (10c): The VTC versus the distance between the Surface and the Camera for texture D9



Relative Range = 900 mm

Relative Range = 550 mm

Relative Range = 400 mm

Relative Range = 280 mm

Relative Range = 200 mm

Figure (11a): Image sequence for Texture D12

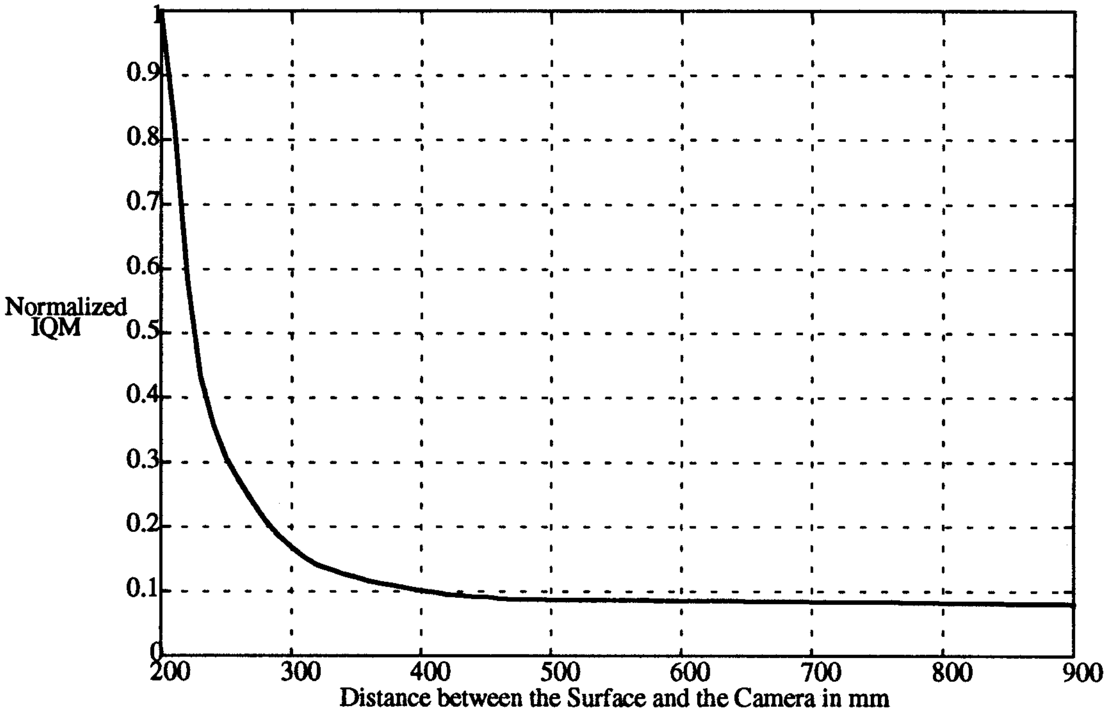


Figure (11b): Normalized IQM versus the distance between the Surface and the Camera for texture D12

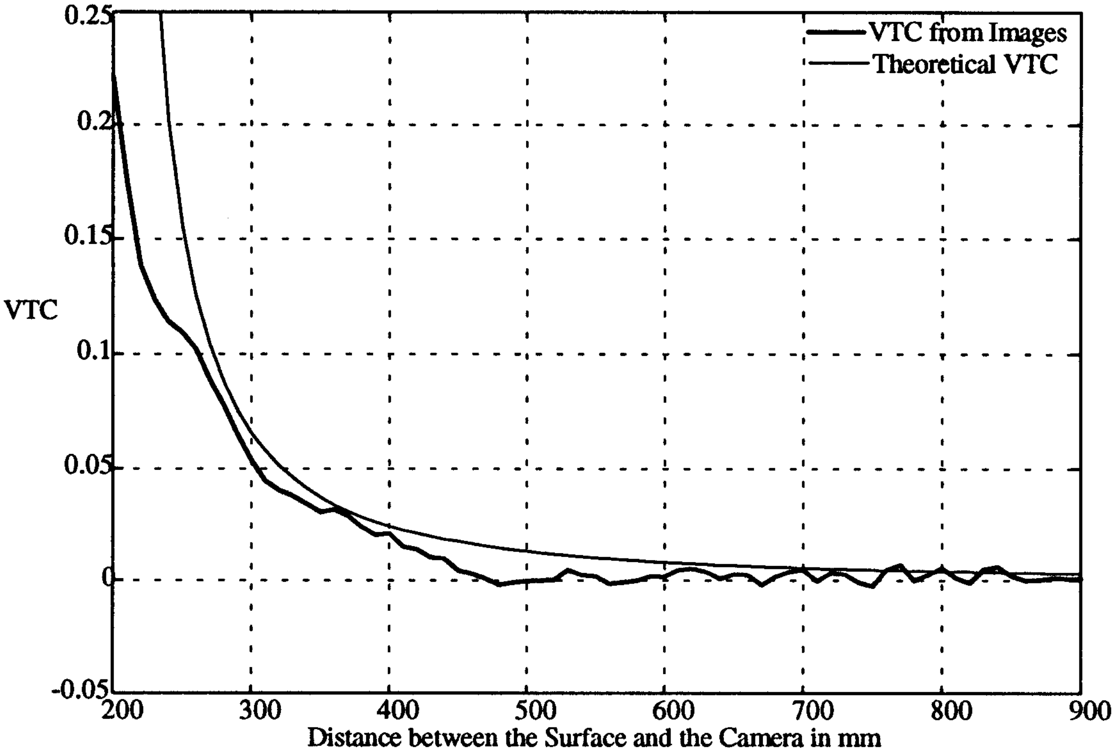
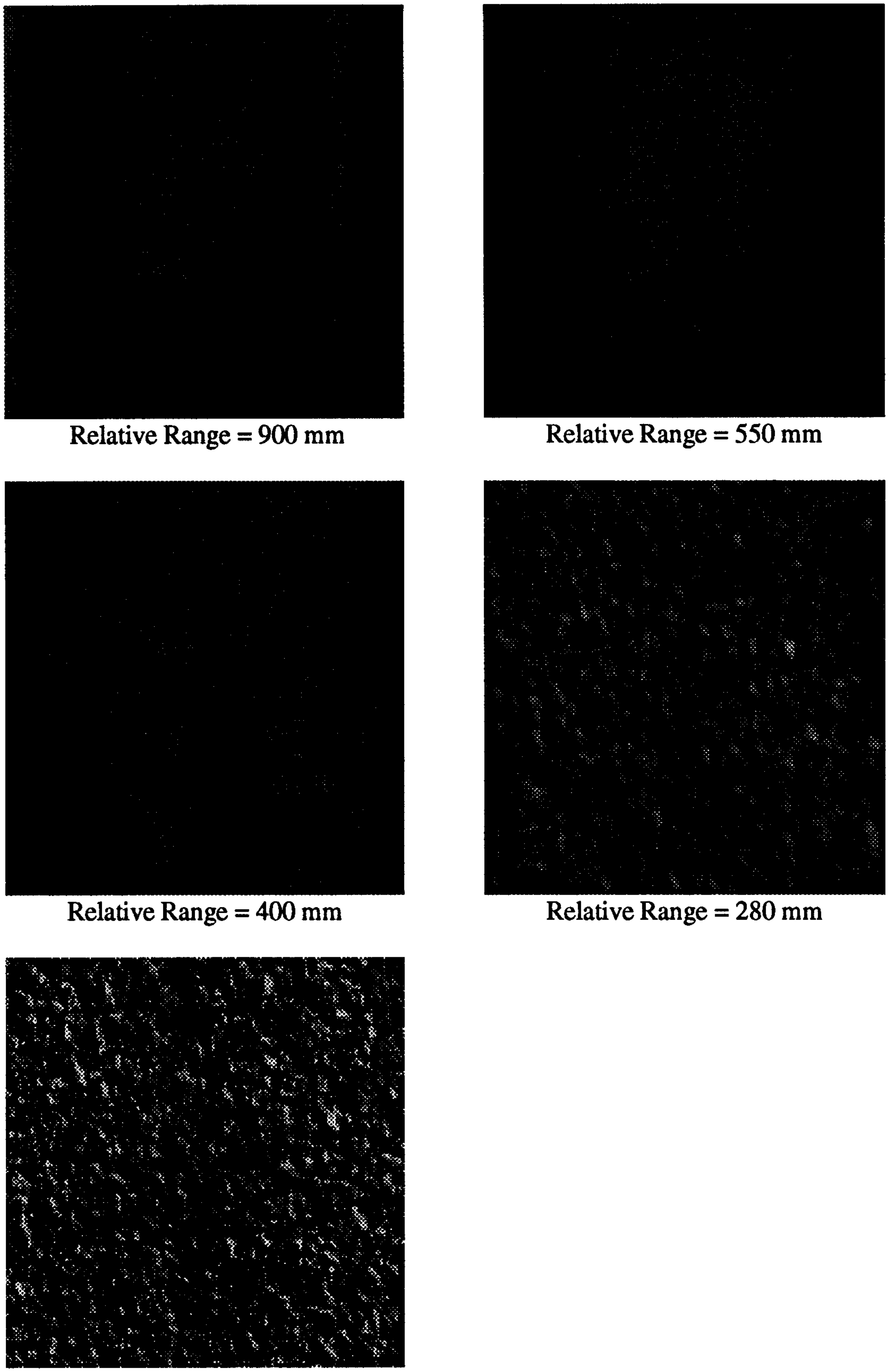


Figure (11c): The VTC versus the distance between the Surface and the Camera for texture D12



Relative Range = 900 mm

Relative Range = 550 mm

Relative Range = 400 mm

Relative Range = 280 mm

Relative Range = 200 mm

Figure (12a): Image Sequence for Texture D4

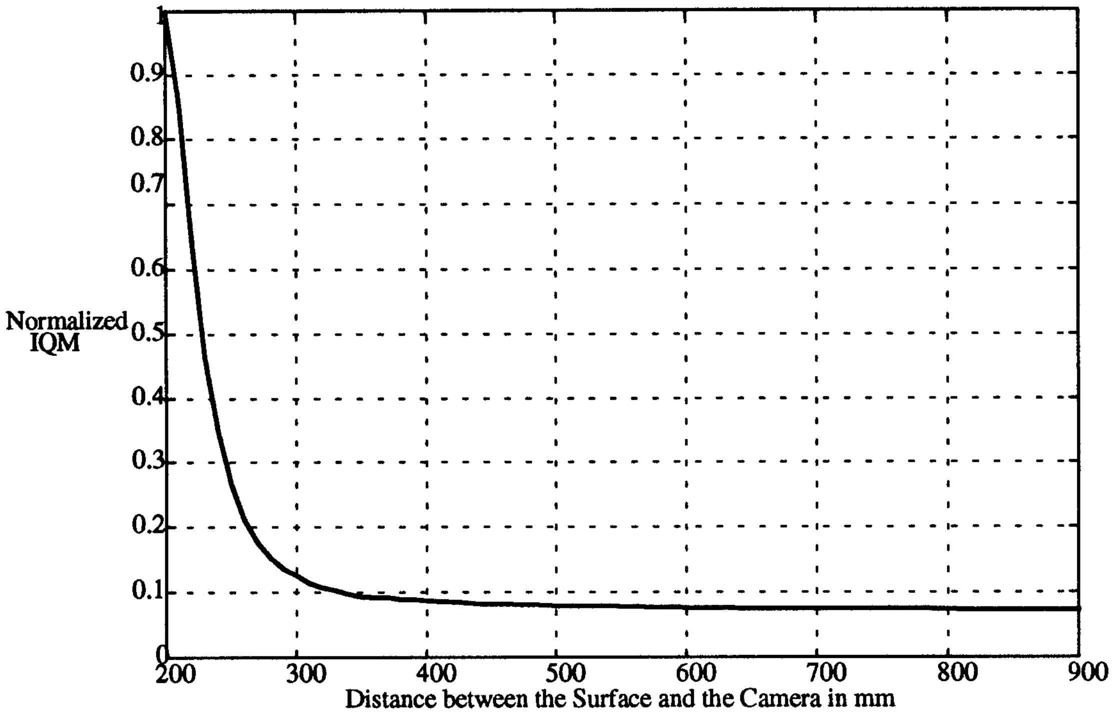


Figure (12b): Normalized IQM versus the distance between the Surface and the Camera for texture D4

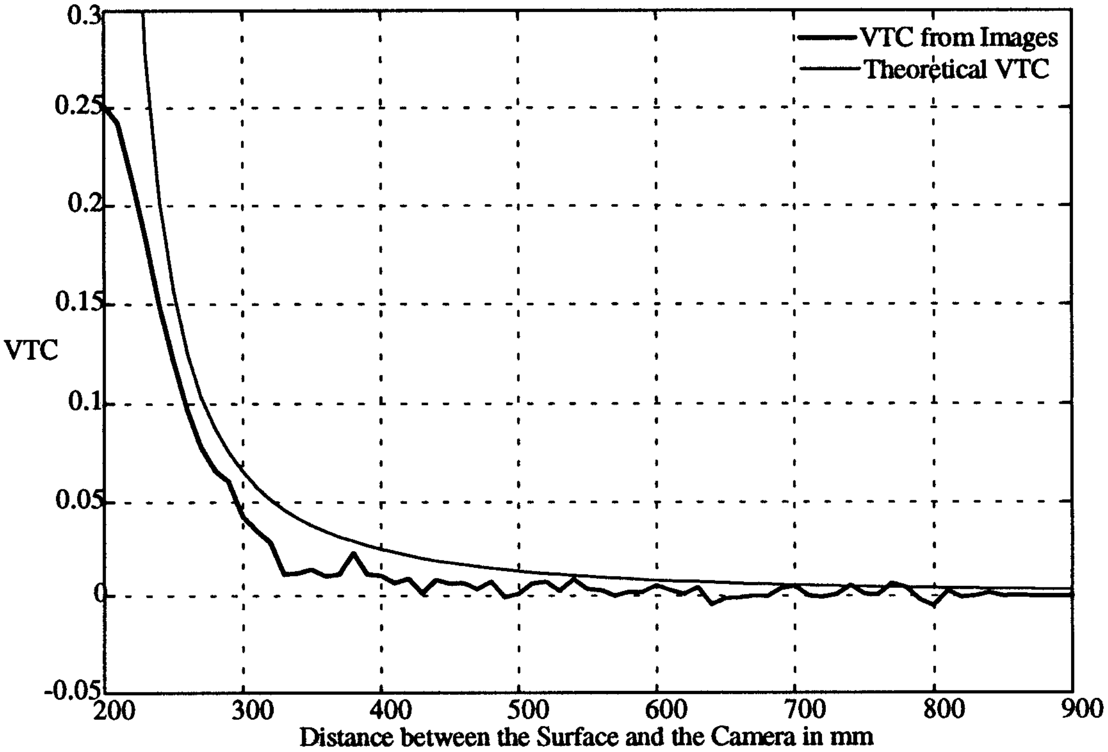
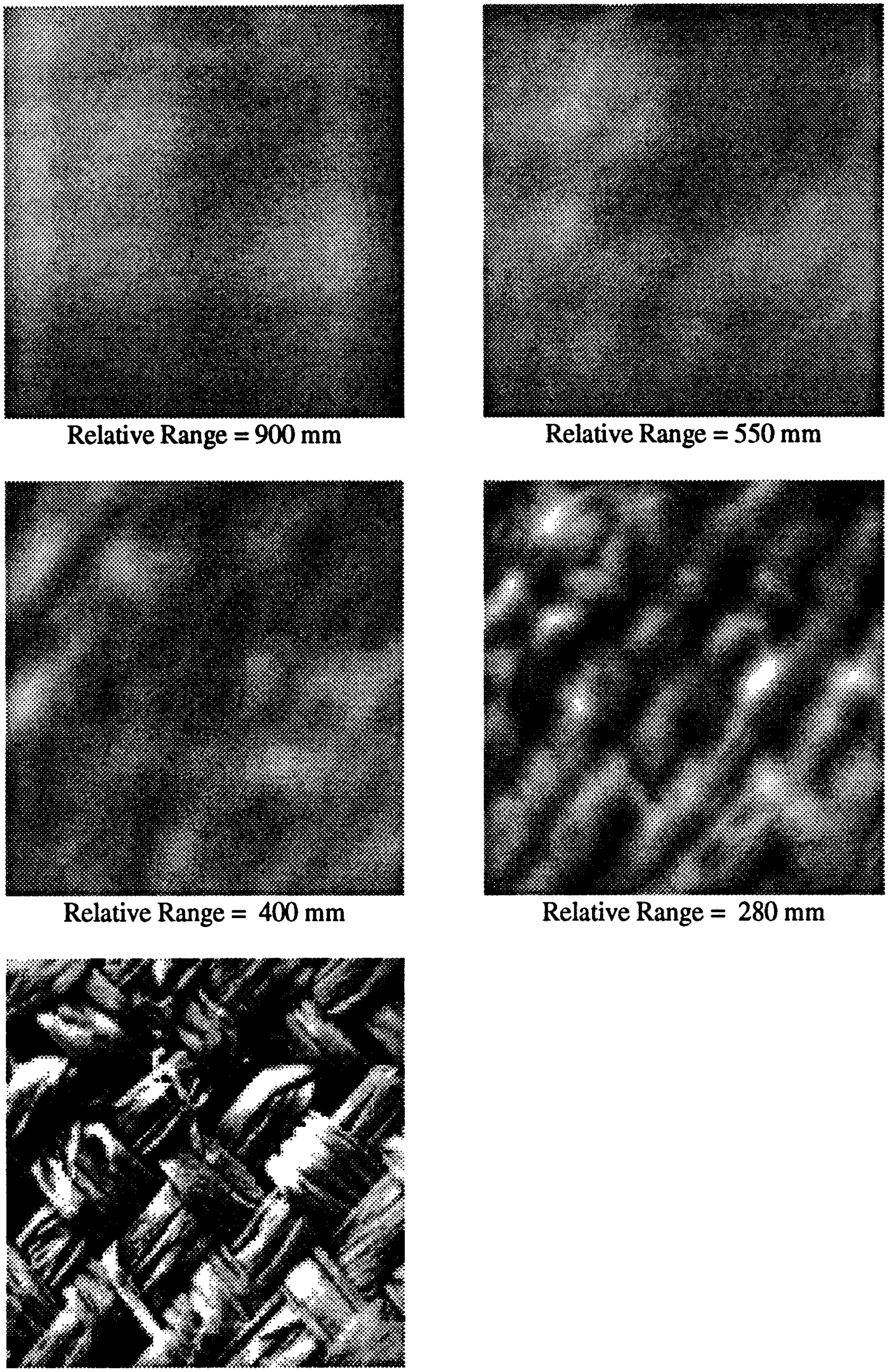


Figure (12c): The VTC versus the distance between the Surface and the Camera for texture D4



Relative Range = 900 mm

Relative Range = 550 mm

Relative Range = 400 mm

Relative Range = 280 mm

Relative Range = 200 mm

Figure (13a): Image Sequence for Texture D18

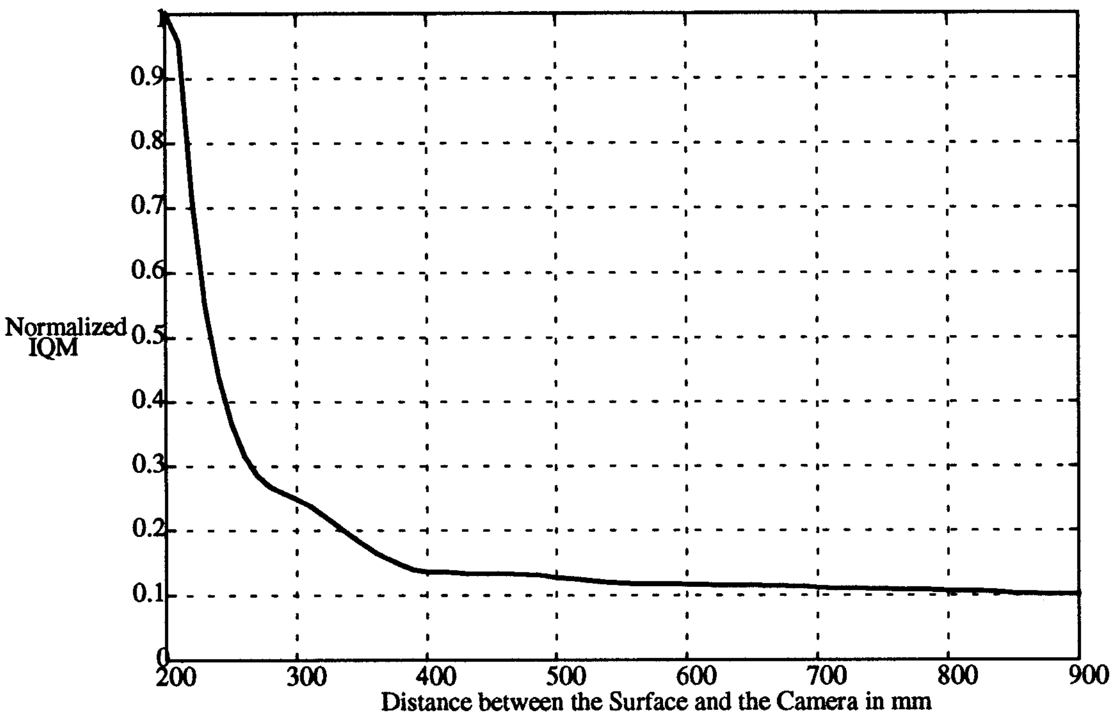


Figure (13b): Normalized IQM versus the distance between the Surface and the Camera for texture D18

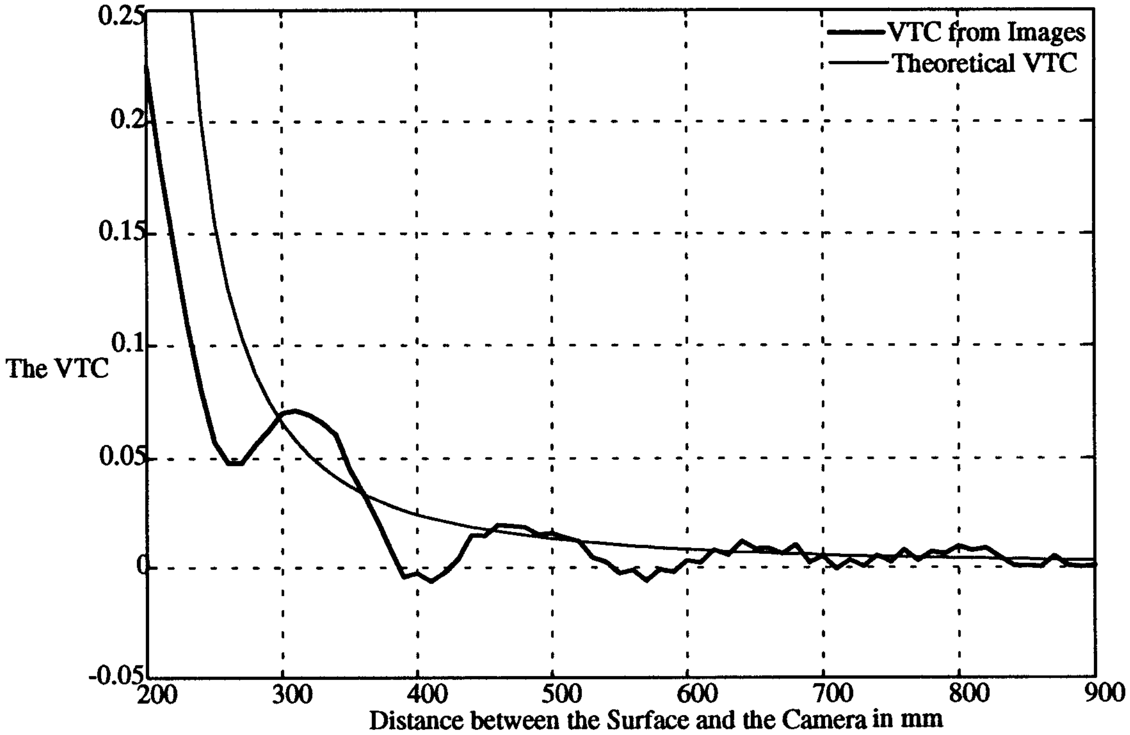
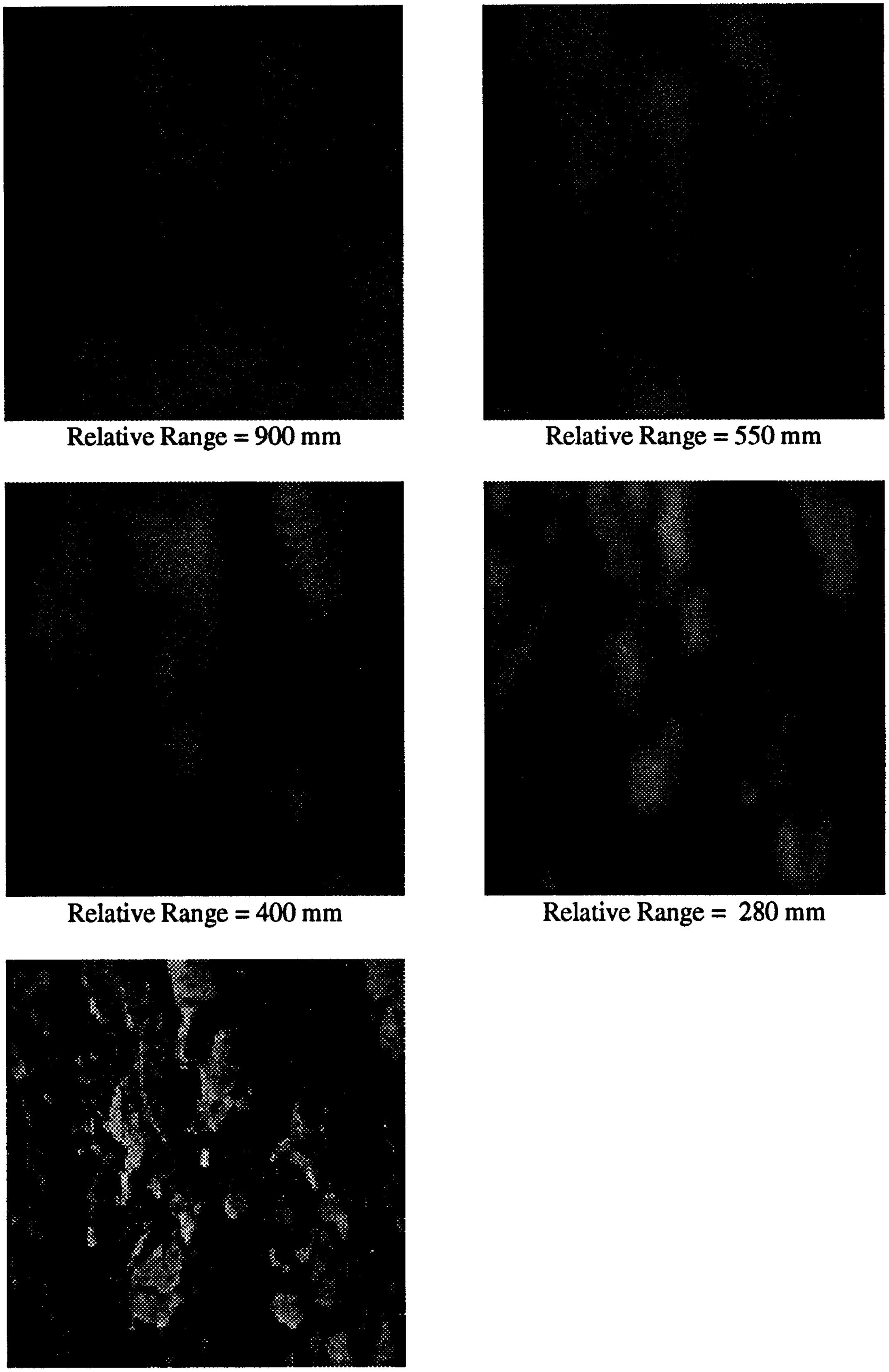


Figure (13c): The VTC versus the distance between the Surface and the Camera for texture D18



Relative Range = 900 mm

Relative Range = 550 mm

Relative Range = 400 mm

Relative Range = 280 mm

Relative Range = 200 mm

Figure (14a): Image Sequence for Texture D13

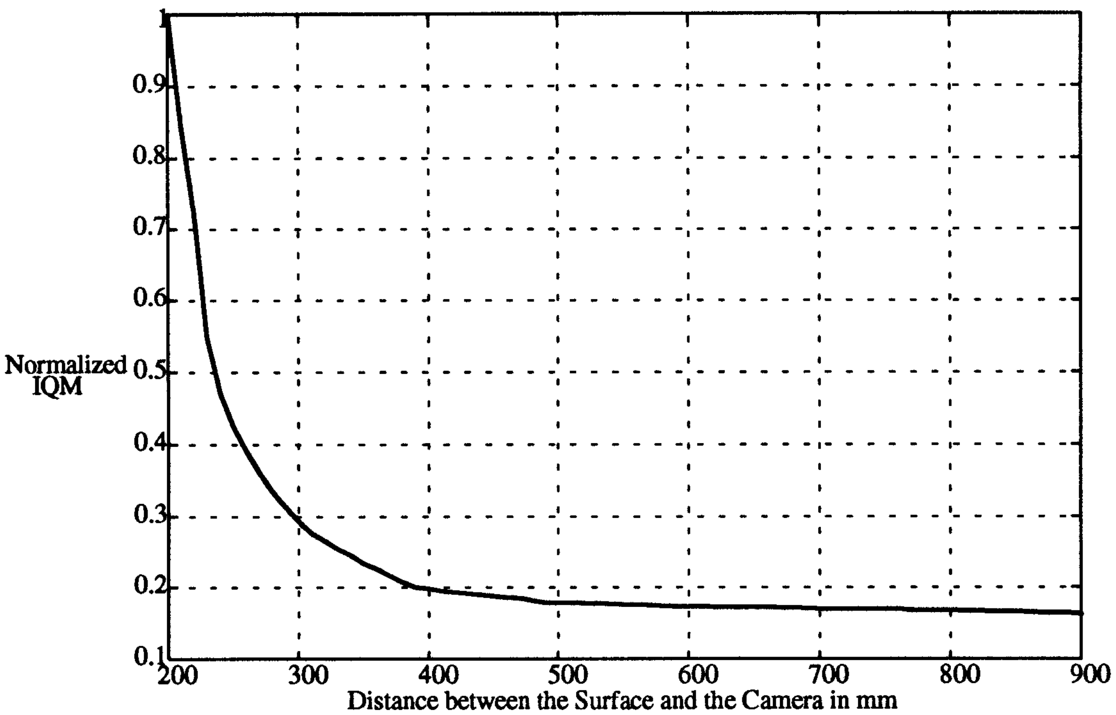


Figure (14b): Normalized IQM versus the distance between the Surface and the Camera for texture D13

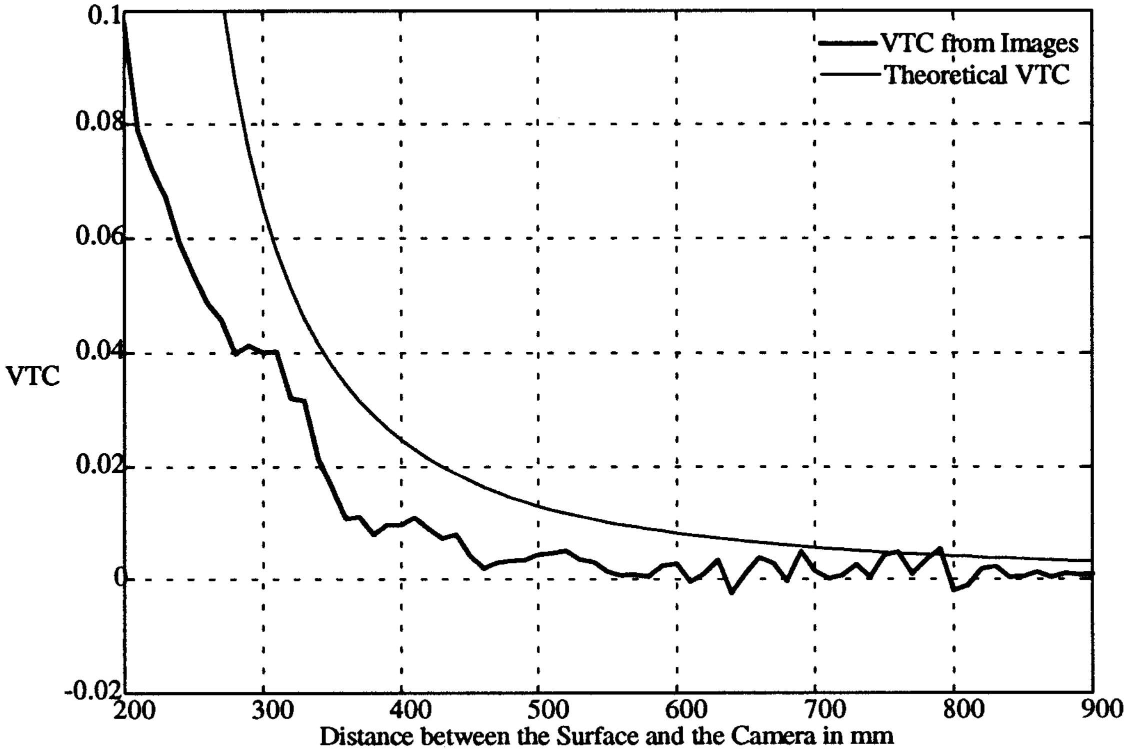
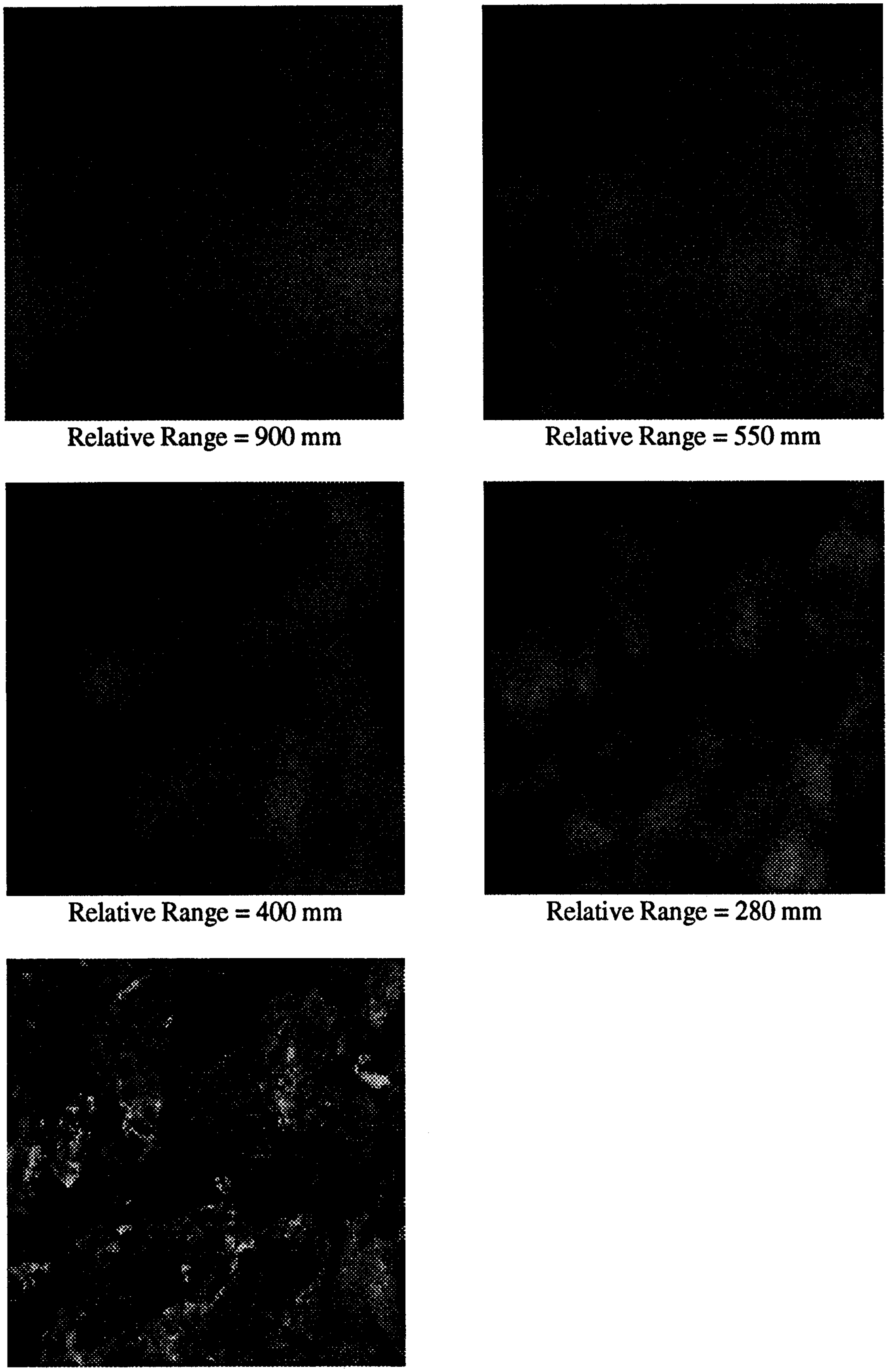


Figure (14c): The VTC versus the distance between the Surface and the Camera for texture D13



Relative Range = 900 mm

Relative Range = 550 mm

Relative Range = 400 mm

Relative Range = 280 mm

Relative Range = 200 mm

Figure (15a): Image Sequence for Texture D98

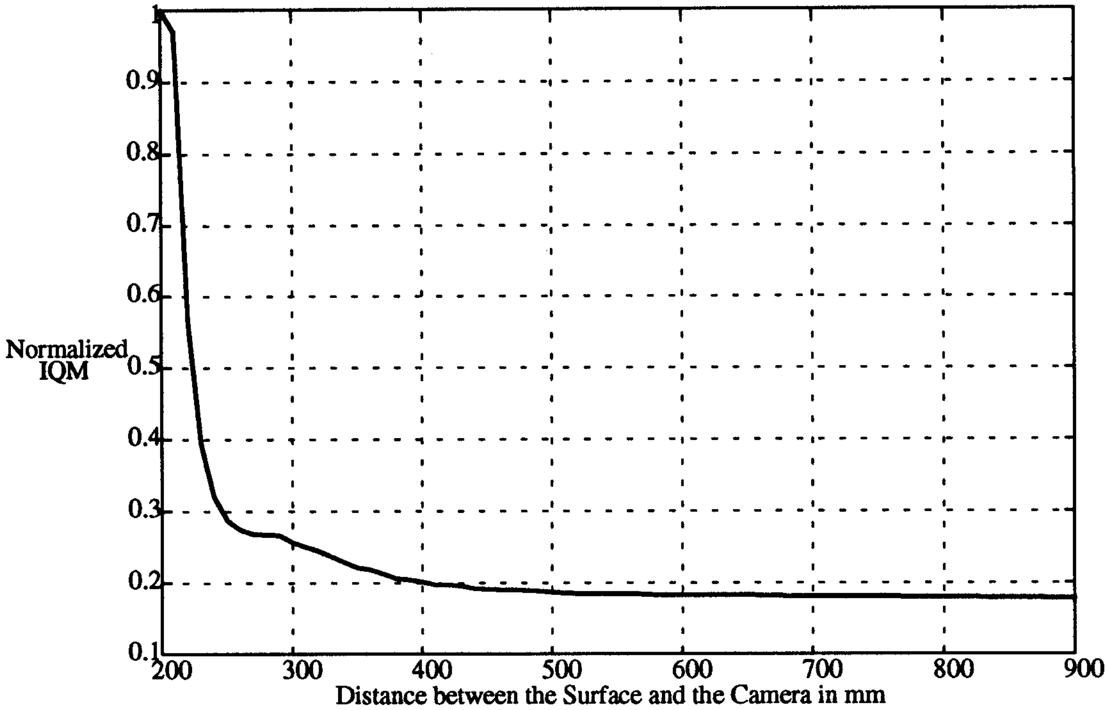


Figure (15b): Normalized IQM versus the distance between the Surface and the Camera for texture D98

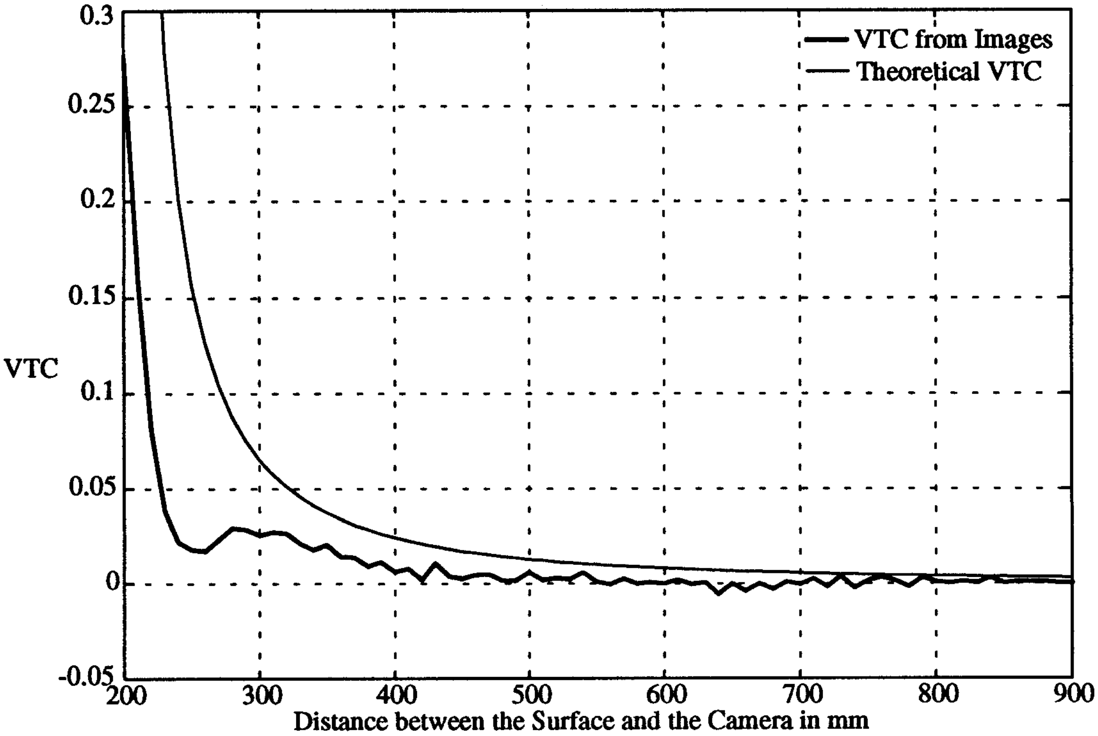
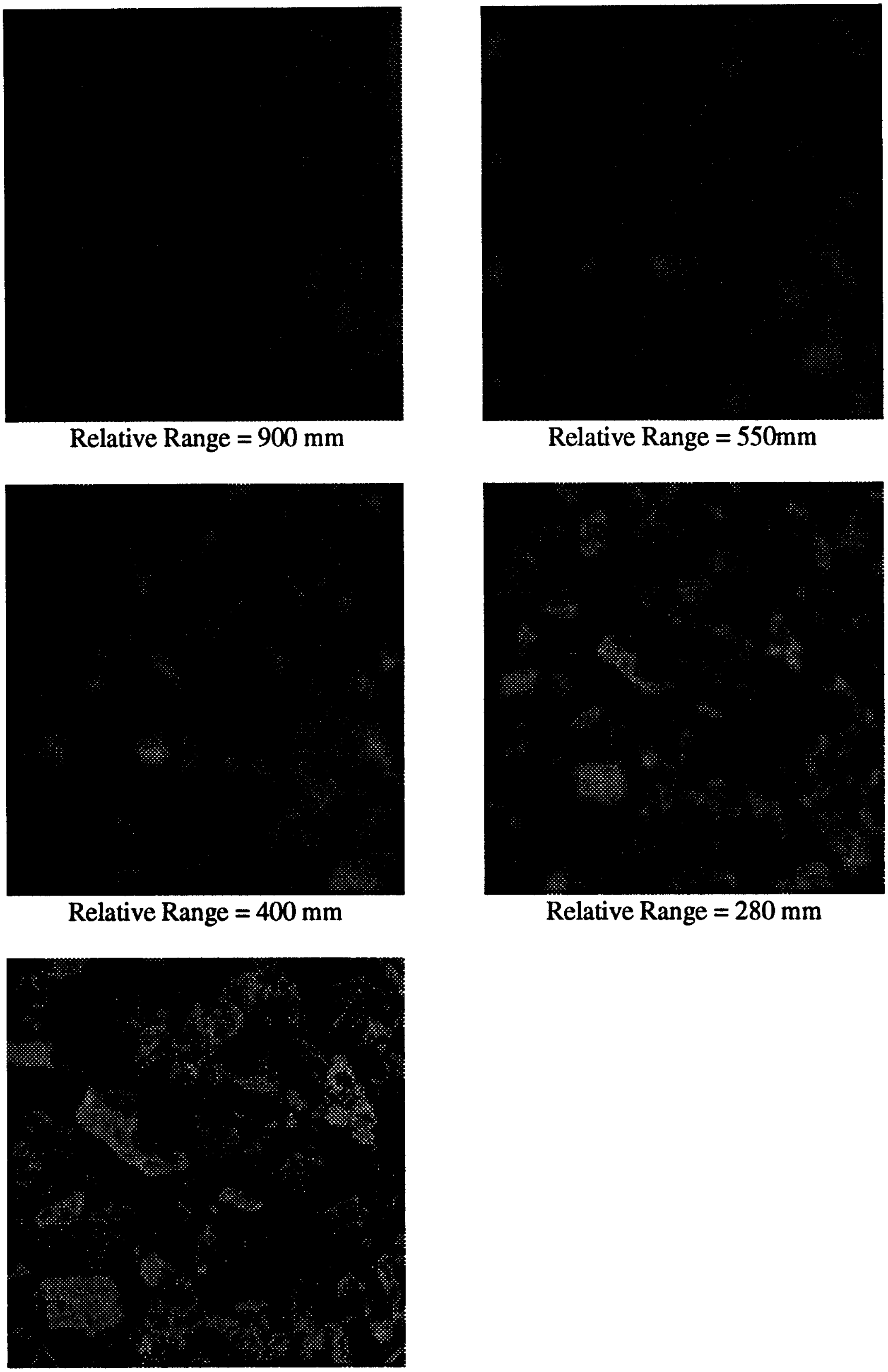


Figure (15c): The VTC versus the distance between the Surface and the Camera for texture D98



Relative Range = 900 mm

Relative Range = 550mm

Relative Range = 400 mm

Relative Range = 280 mm

Relative Range = 200 mm

Figure (16a): Image Sequence for Texture D5

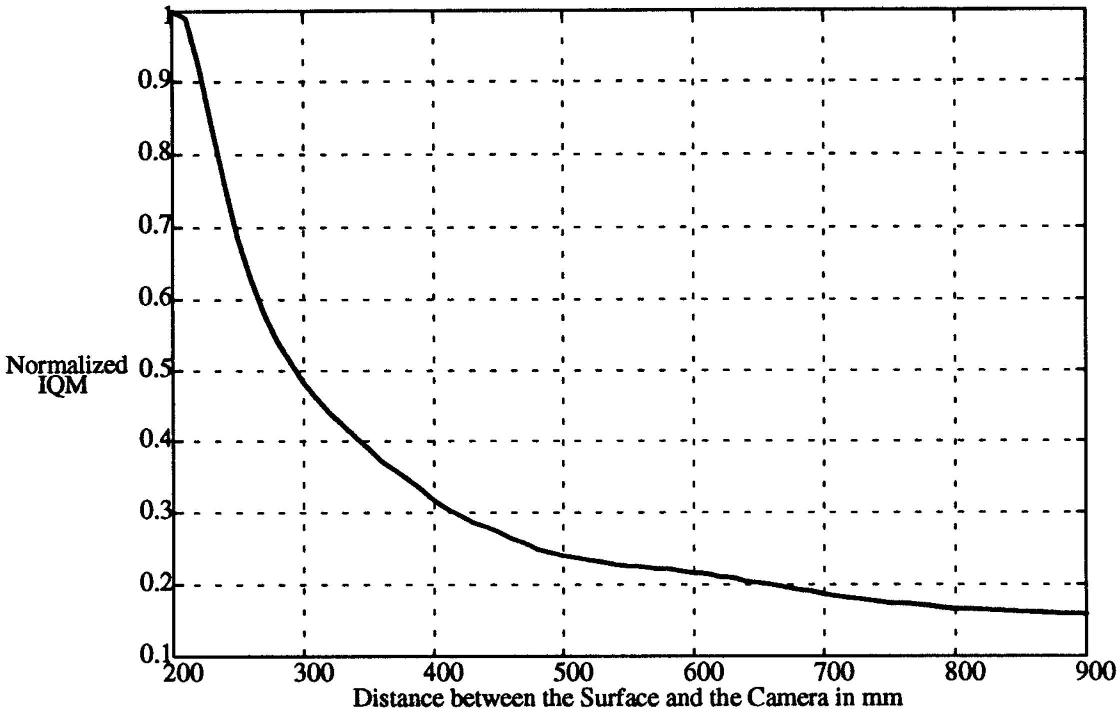


Figure (16b): Normalized IQM versus the distance between the Surface and the Camera for texture D5

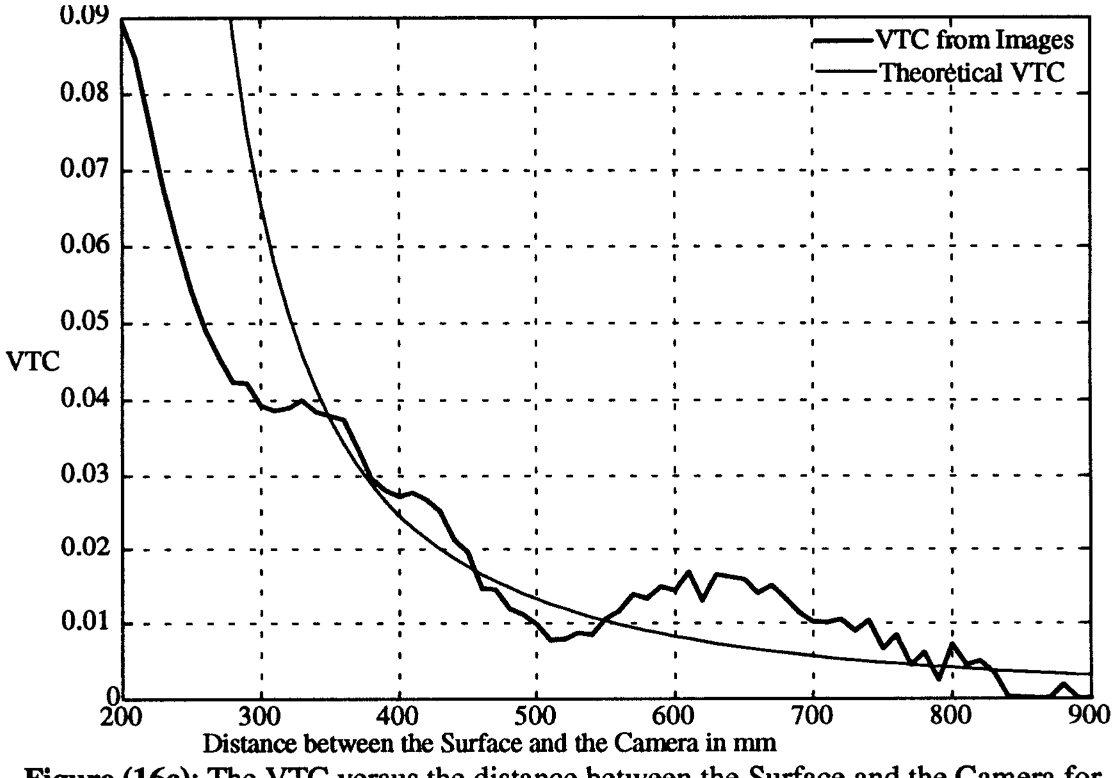
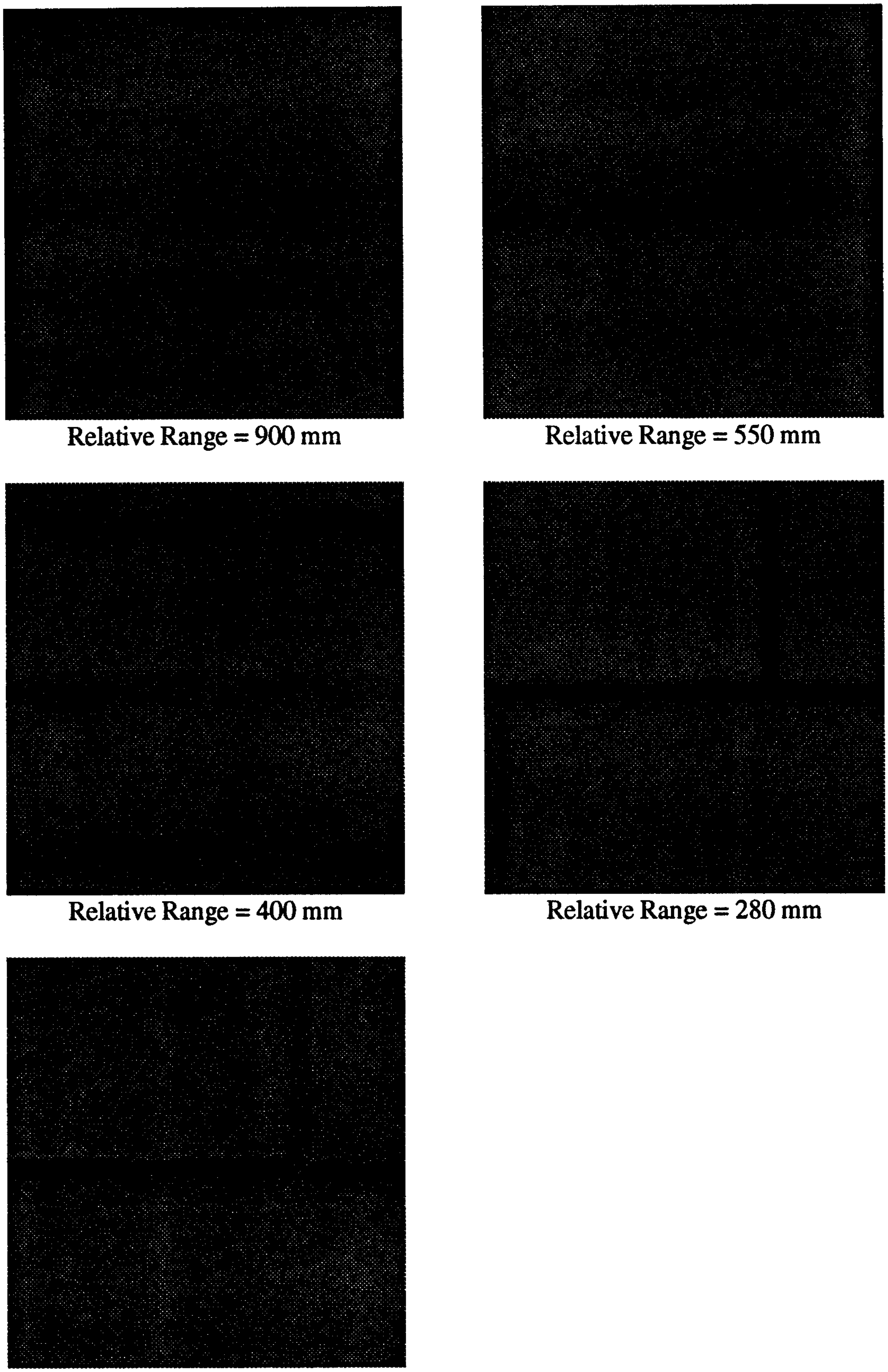


Figure (16c): The VTC versus the distance between the Surface and the Camera for texture D5



Relative Range = 900 mm

Relative Range = 550 mm

Relative Range = 400 mm

Relative Range = 280 mm

Relative Range = 200 mm

Figure (17a): Image Sequence for Texture D25

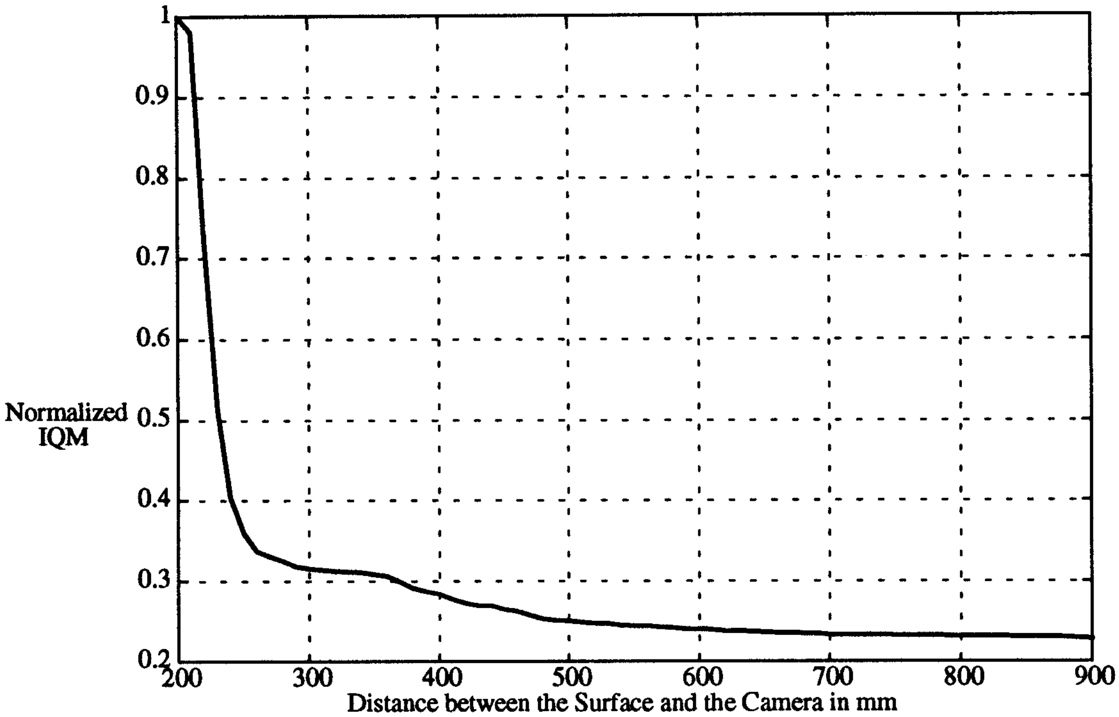


Figure (17b): Normalized IQM versus the distance between the Surface and the Camera for texture D25

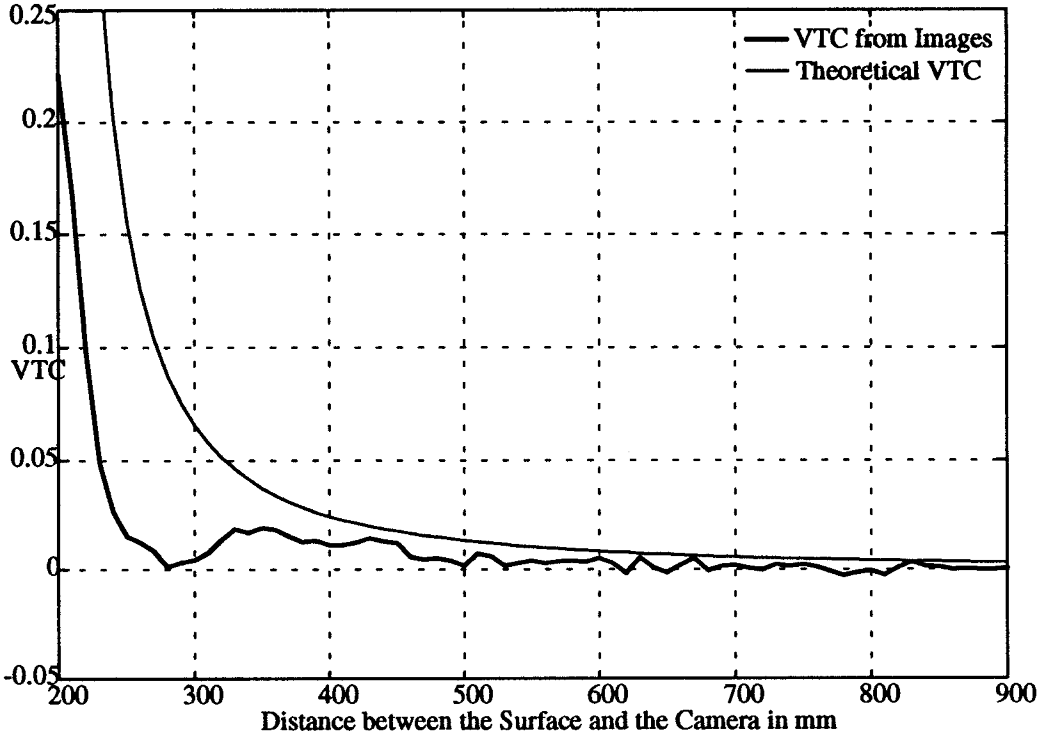
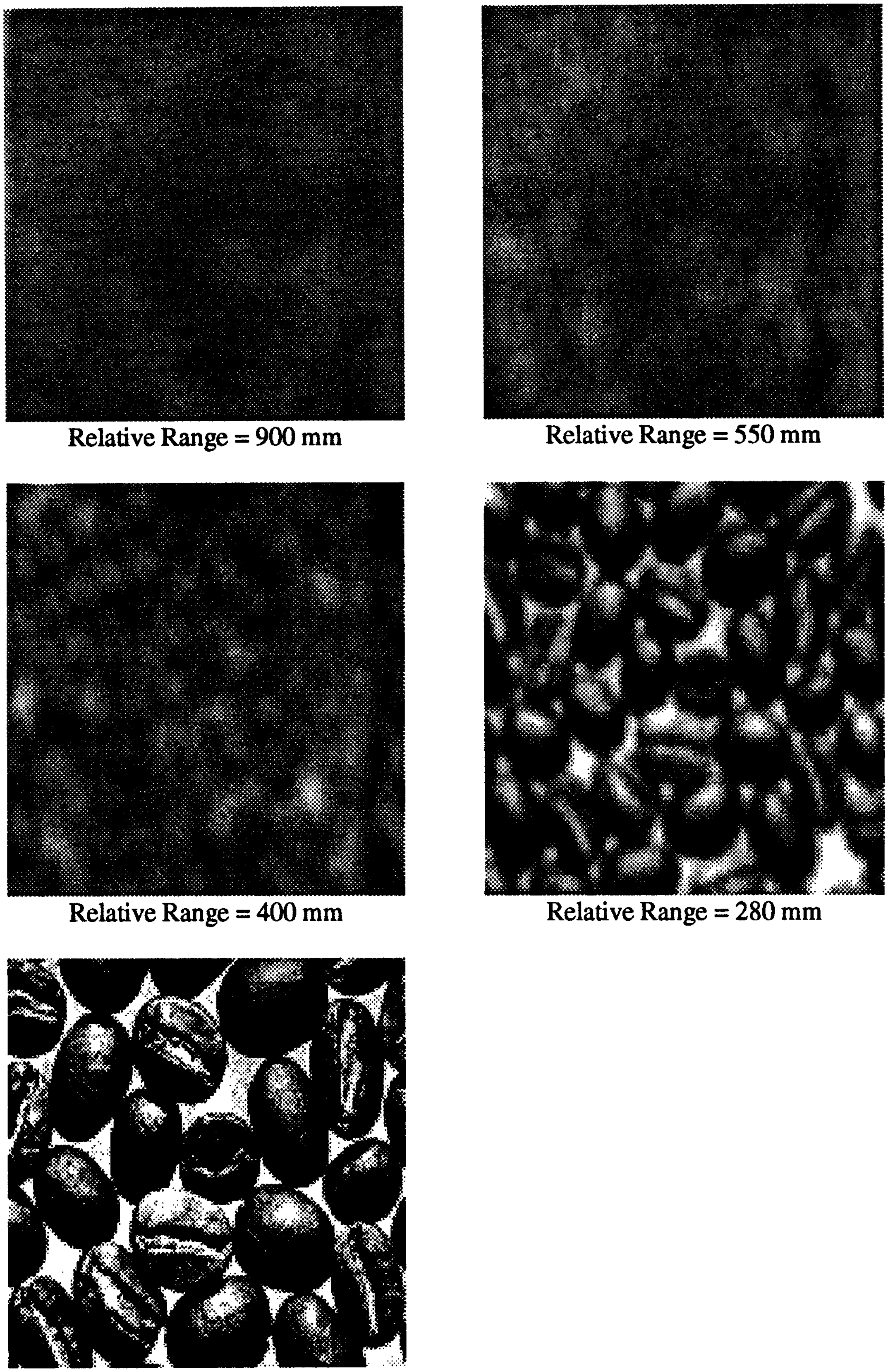


Figure (17c): The VTC versus the distance between the Surface and the Camera for texture D25



Relative Range = 900 mm

Relative Range = 550 mm

Relative Range = 400 mm

Relative Range = 280 mm

Relative Range = 200 mm

Figure (18a): Image Sequence for Texture D74

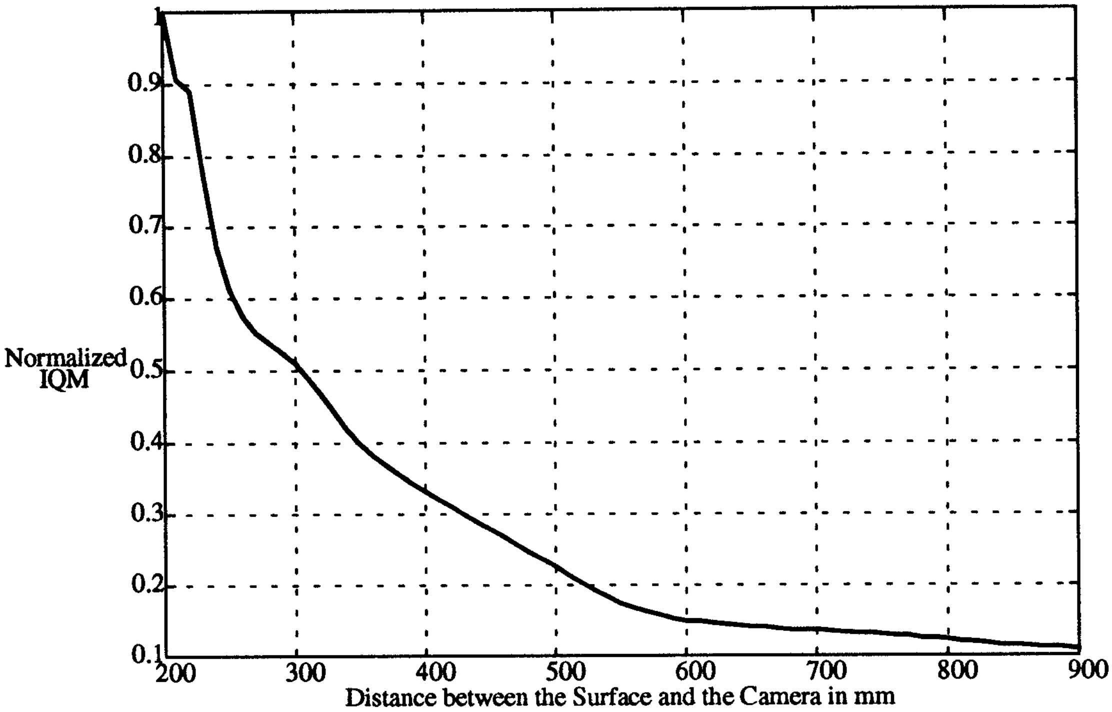


Figure (18b): Normalized IQM versus the distance between the Surface and the Camera for texture D74

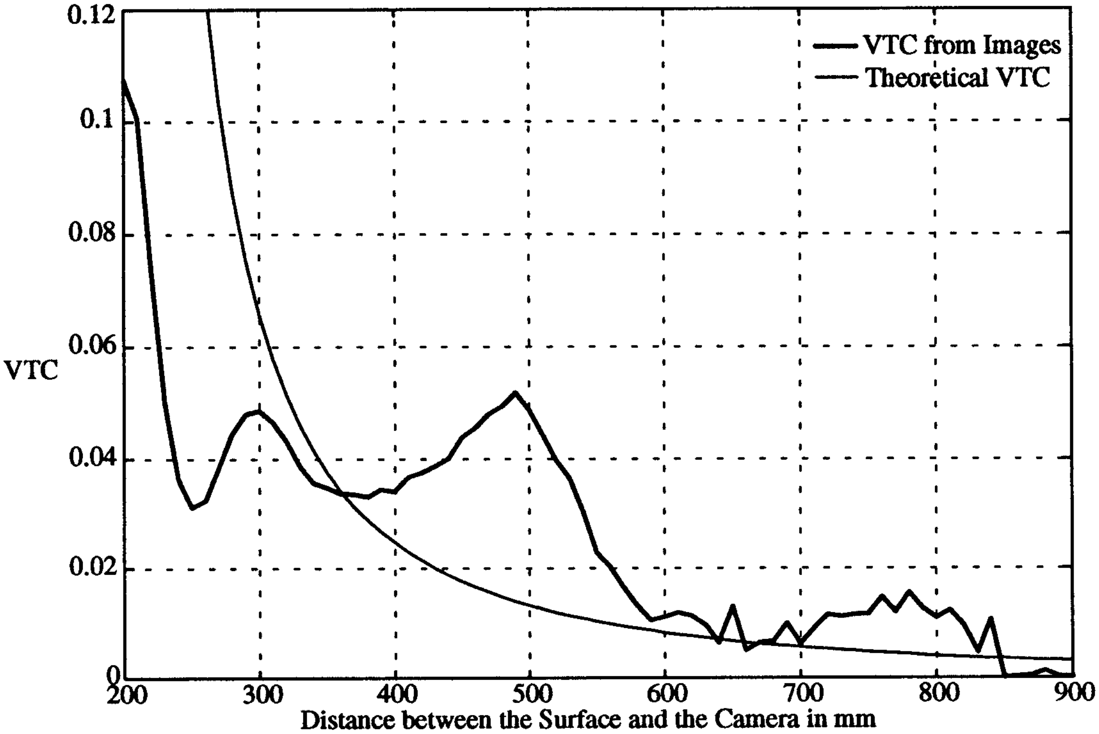
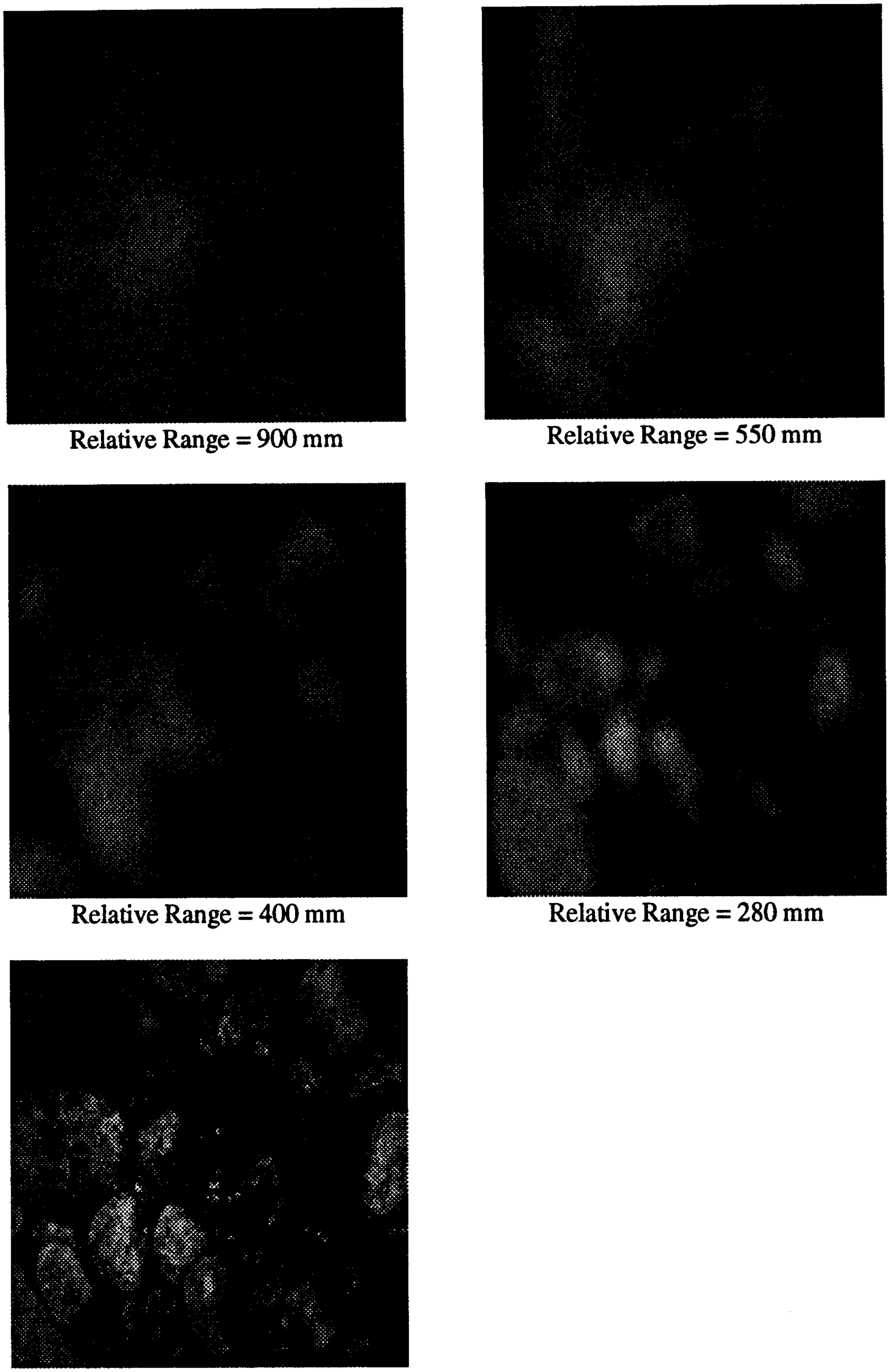


Figure (18c): The VTC versus the distance between the Surface and the Camera for texture D110



Relative Range = 900 mm

Relative Range = 550 mm

Relative Range = 400 mm

Relative Range = 280 mm

Relative Range = 200 mm

Figure (19a): Image Sequence for Texture D23

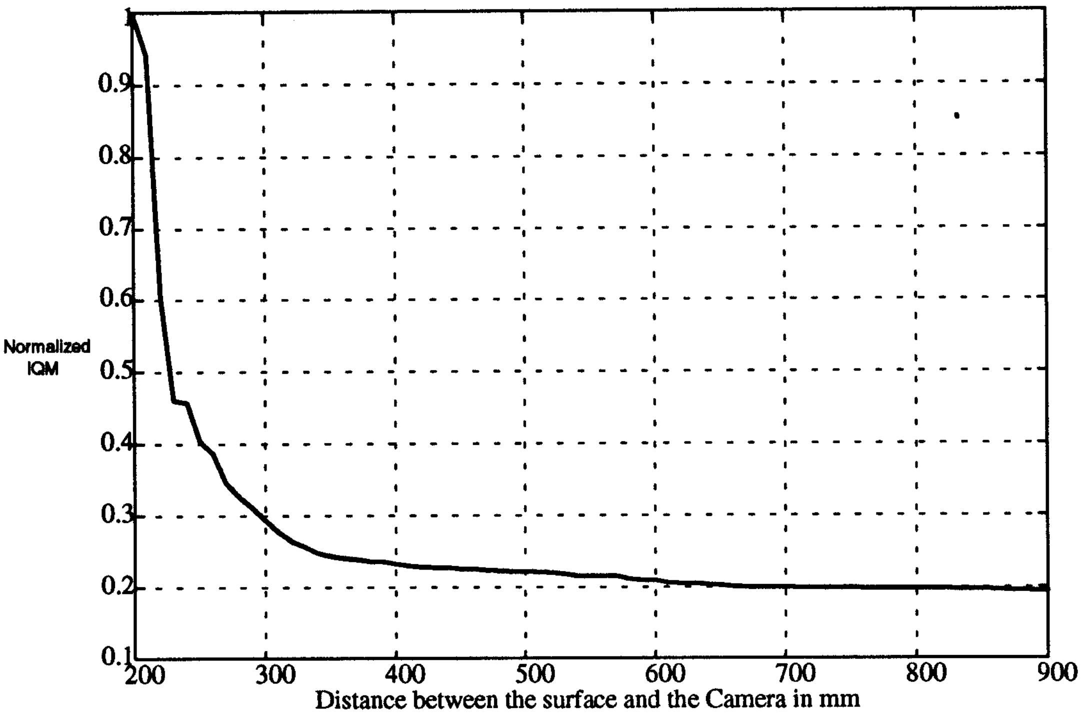


Figure (19b): Normalized IQM versus the Distance between the Surface and the Camera for texture D23

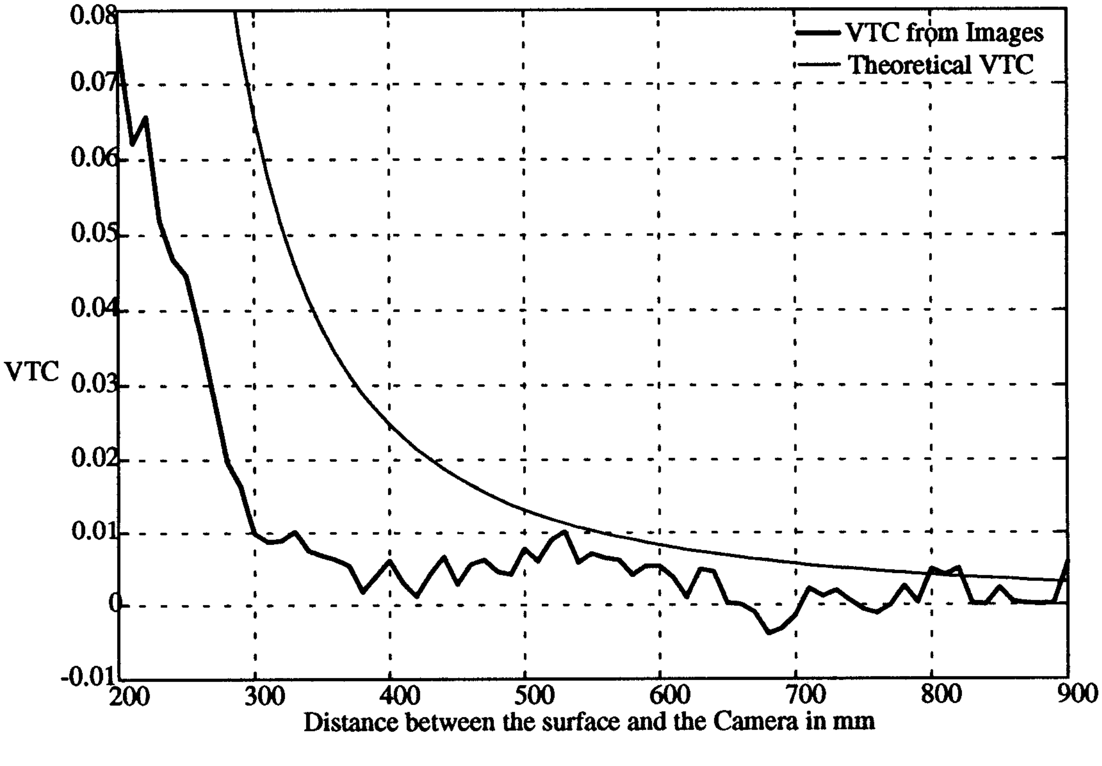
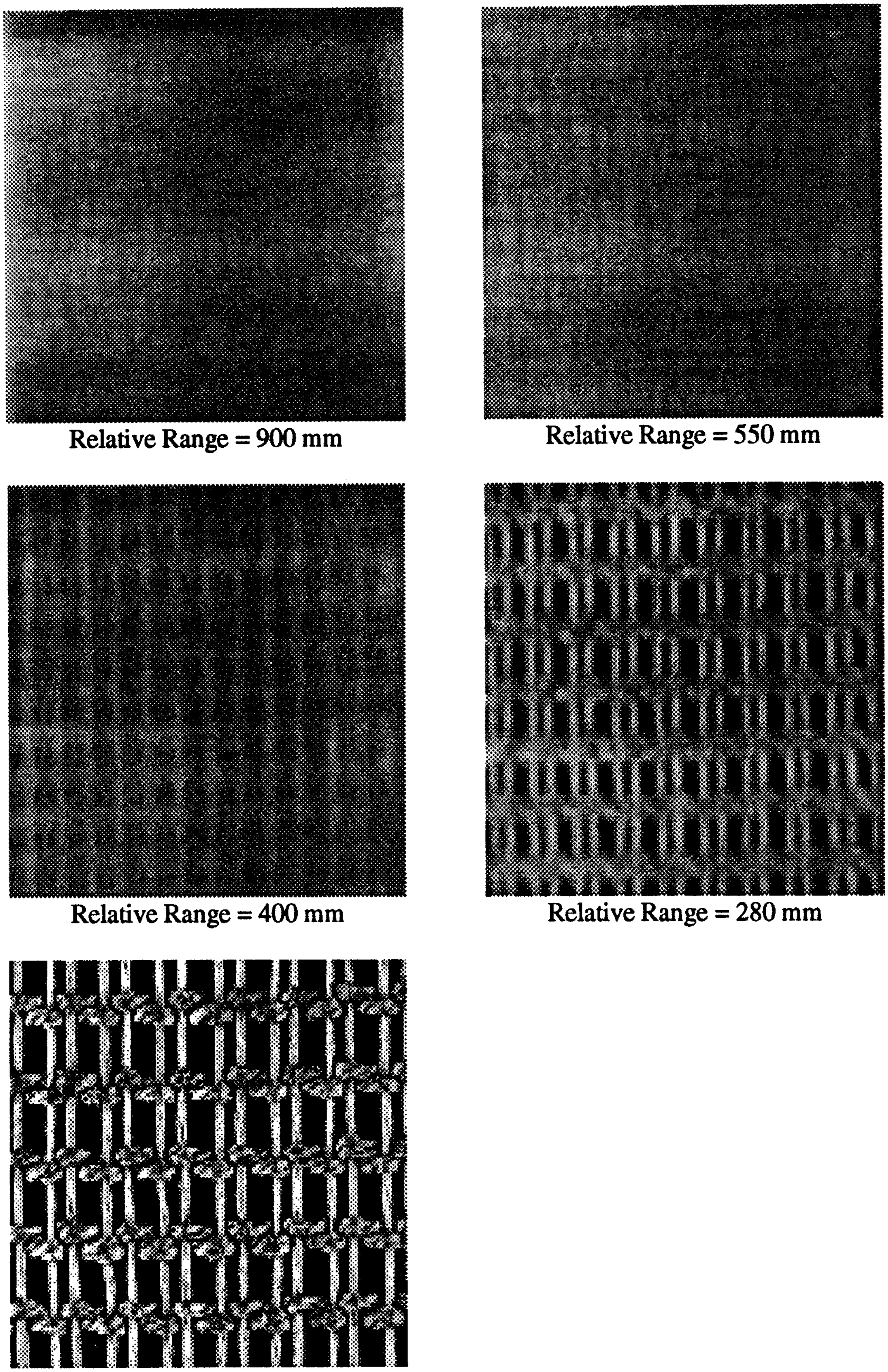


Figure (19c): The VTC versus the Distance between the Surface and the Camera for texture D23



Relative Range = 900 mm

Relative Range = 550 mm

Relative Range = 400 mm

Relative Range = 280 mm

Relative Range = 200 mm

Figure (20a) : Image Sequence for Texture D20

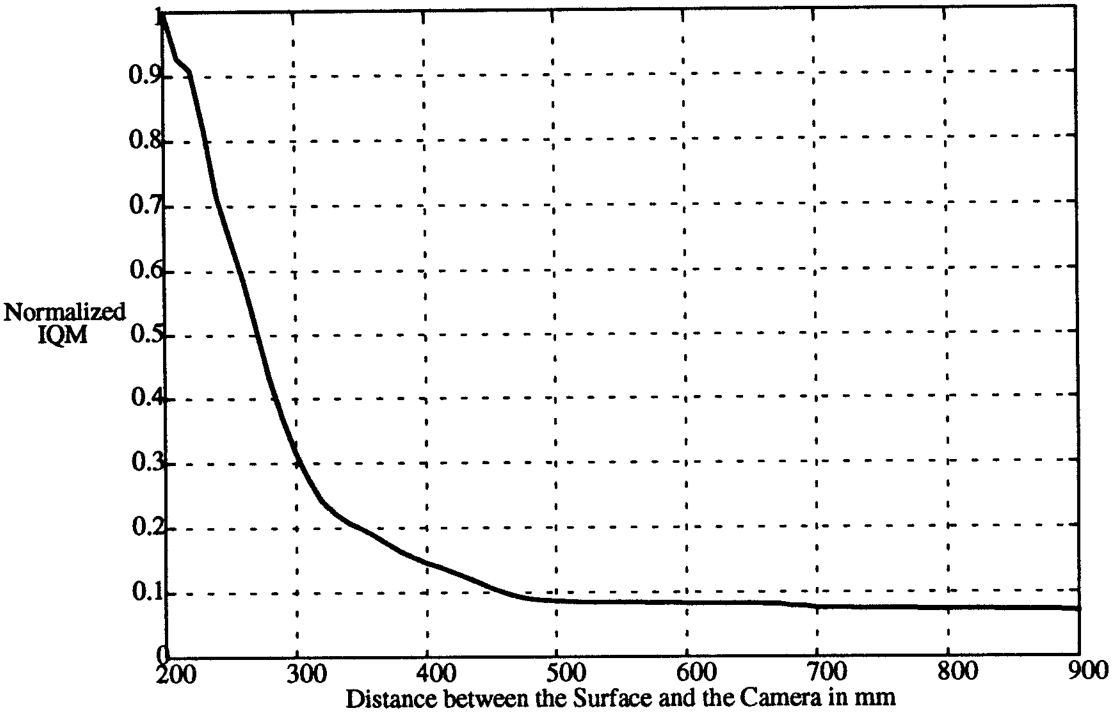


Figure (20b): Normalized IQM versus the Distance between the Surface and the Camera for texture D20

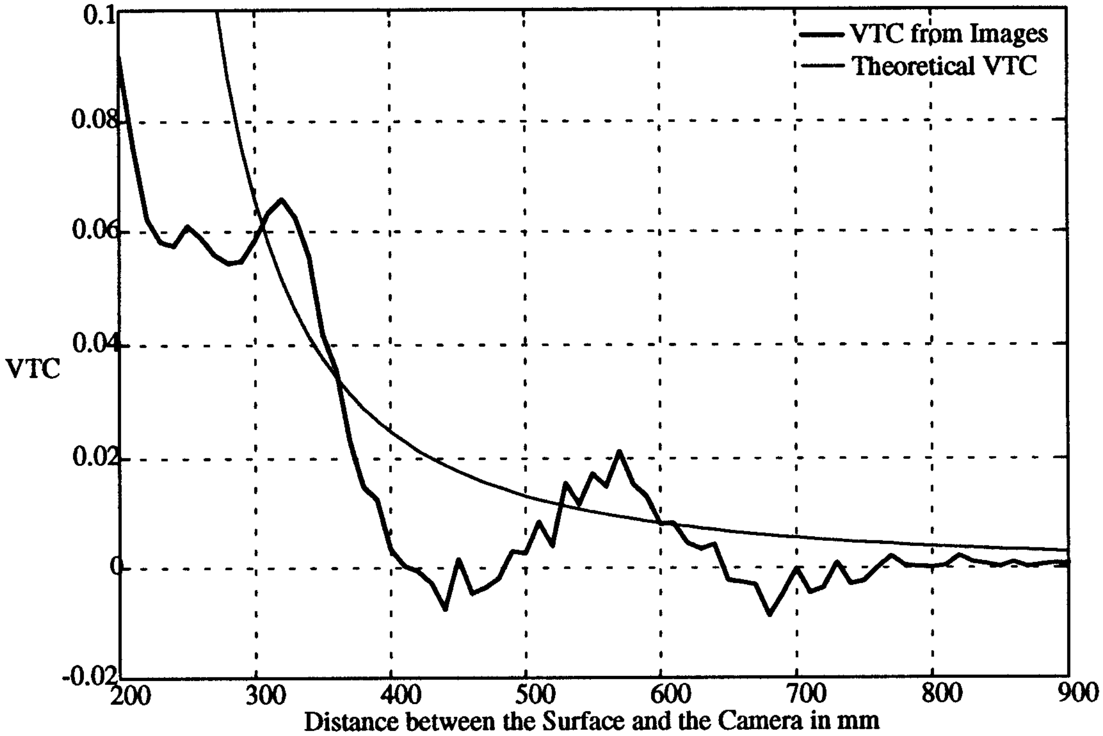


Figure (20c): The VTC versus the Distance between the Surface and the Camera for texture D23

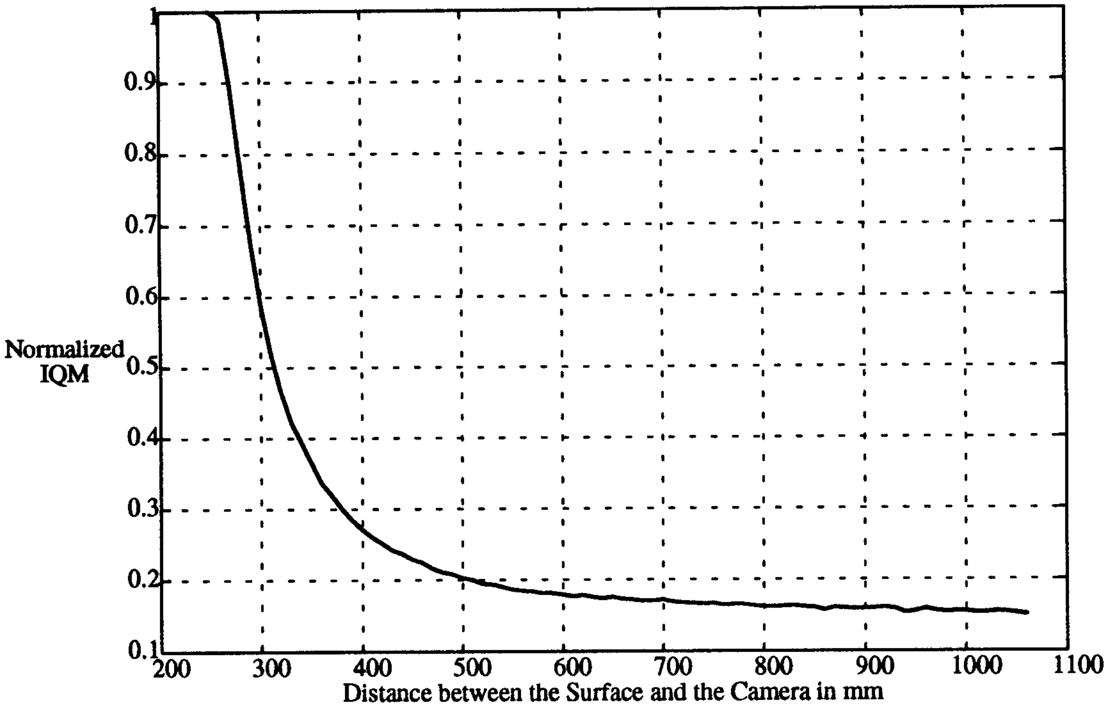


Figure (21a): Normalized IQM versus the Distance between the Surface and the Camera for Texture D110, Minimum Clearance R_0 = 250 mm

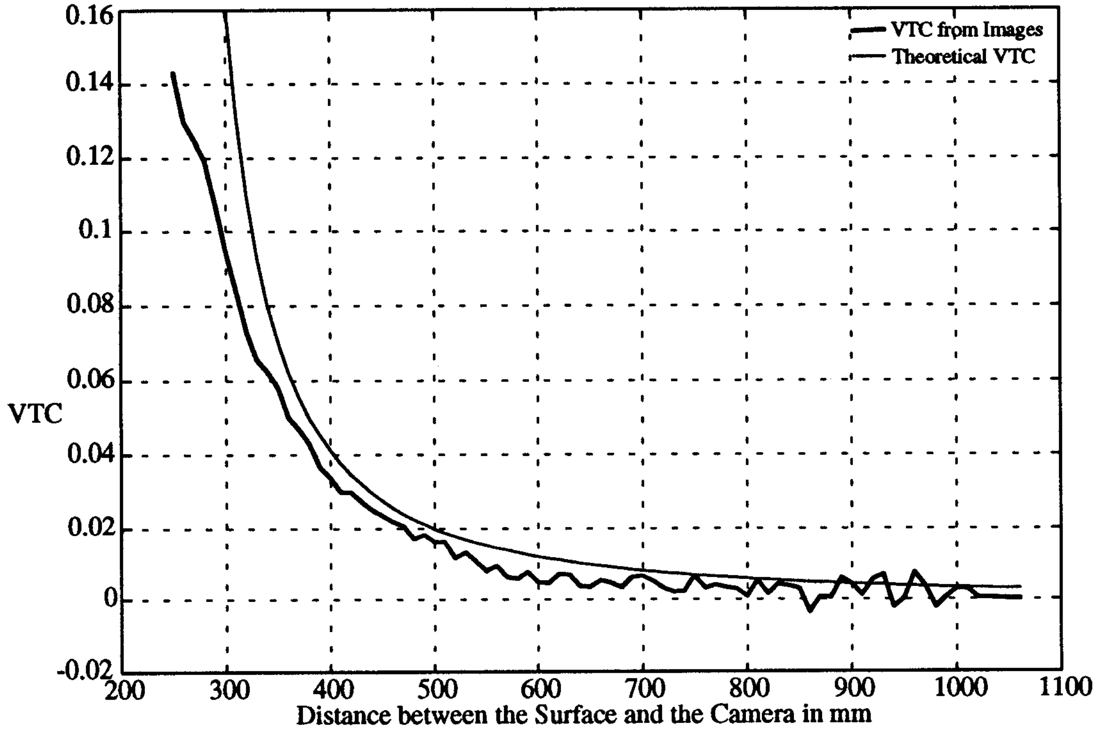


Figure (21b): The VTC versus the Distance between the Surface and the Camera for Texture D110, Minimum Clearance R_0 = 250 mm

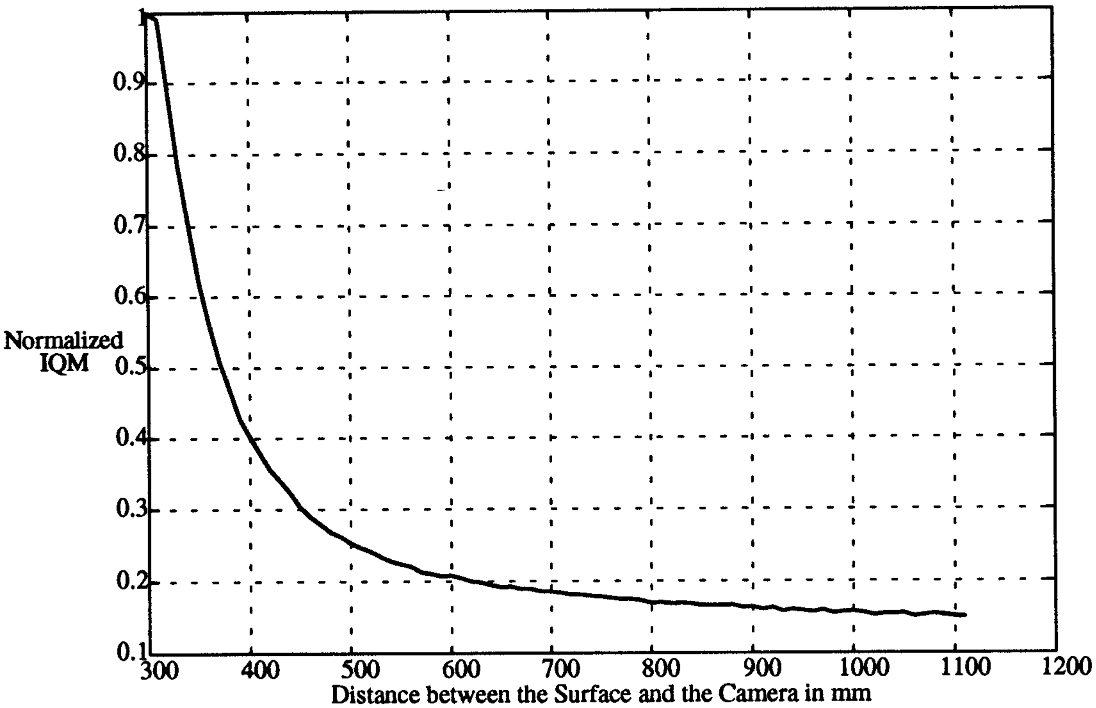


Figure (22a): Normalized IQM versus the Distance between the Surface and the Camera for Texture D110, Minimum Clearance R_0 = 300 mm

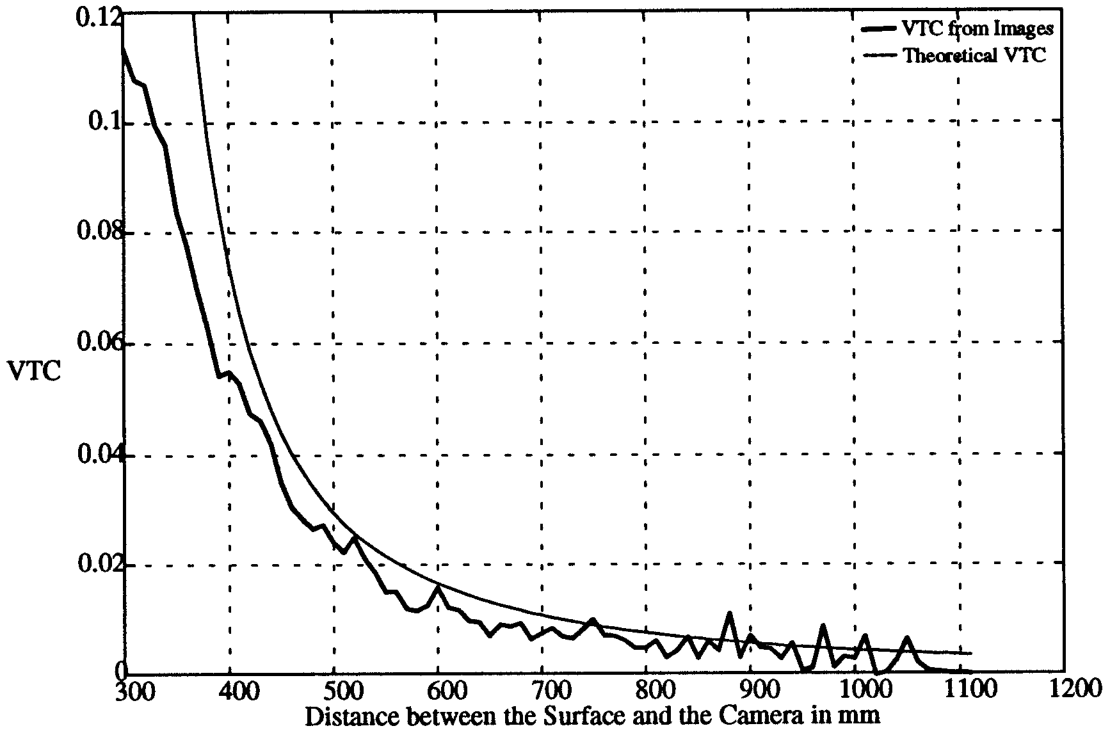


Figure (22b): The VTC versus the Distance between the Surface and the Camera for Texture D110, Minimum Clearance R_0 = 300 mm

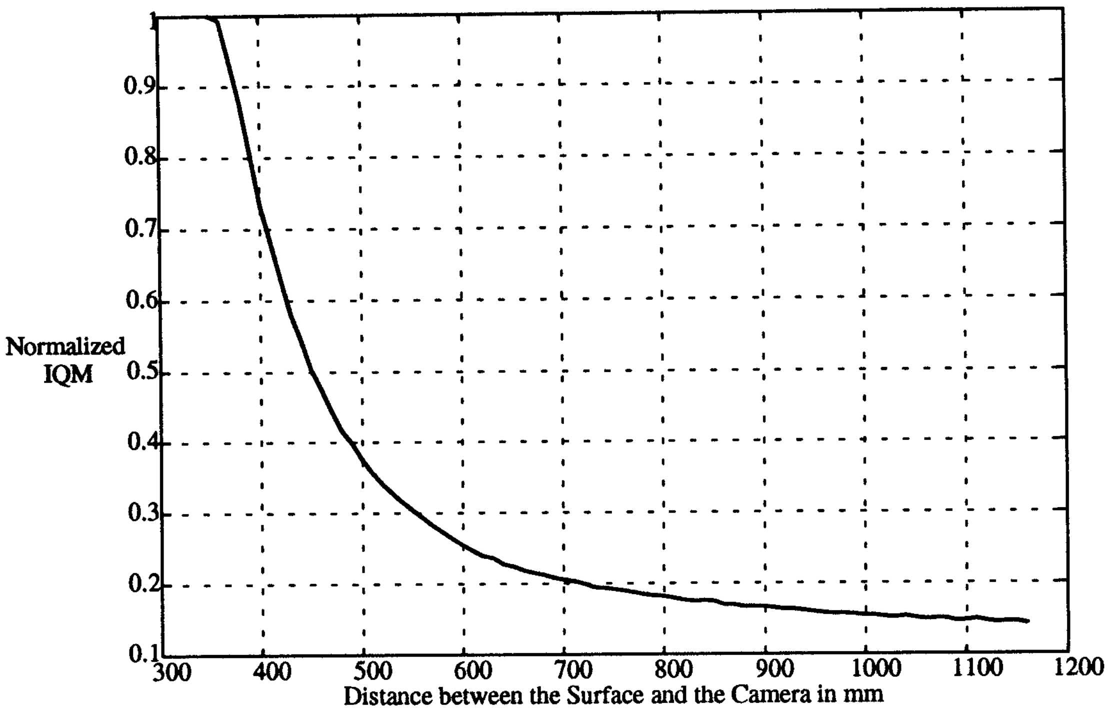


Figure (23a): Normalized IQM versus the Distance between the Surface and the Camera for Texture D110, Minimum Clearance R_0 = 350 mm

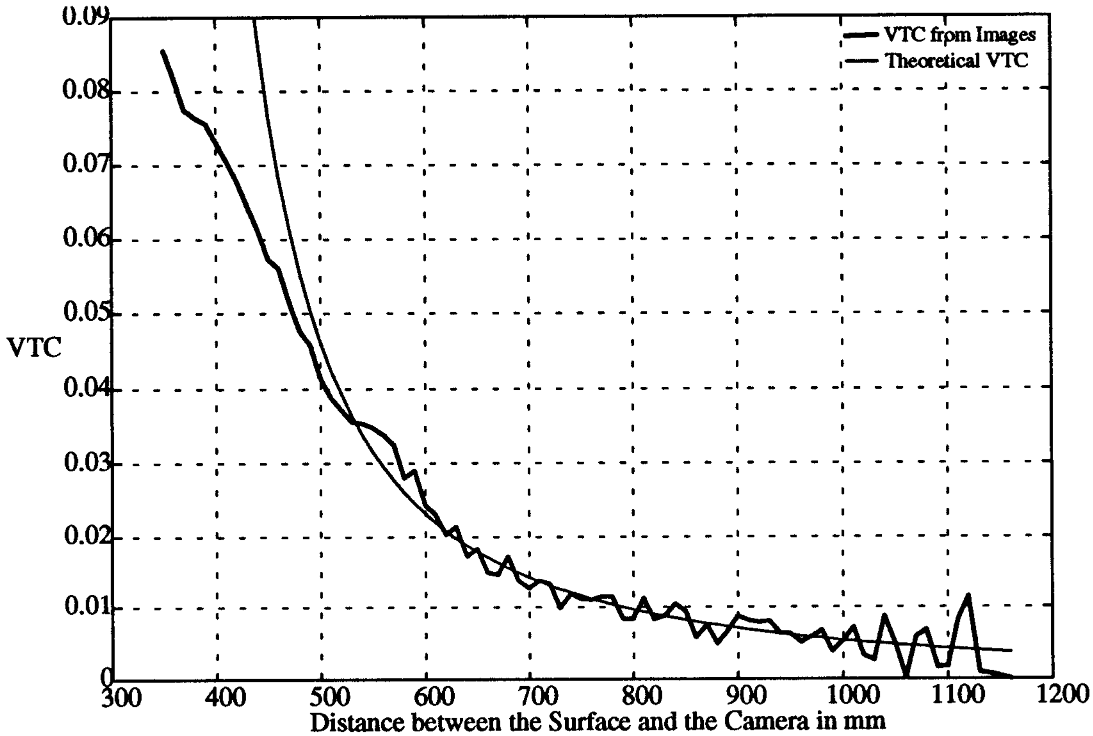


Figure (23b): The VTC versus the Distance between the Surface and the Camera for Texture D110, Minimum Clearance R_0 = 350 mm

6 Conclusions and Future Work

In this paper we present a new visual motion cue, we call the Visual Threat Cue (VTC) that provides a measure for some *relative change in range* between a 3D surface and a *fixated* observer in *motion*. A mathematical definition, a theoretical model as well as a robust practical way to extract it from a sequence of images are presented. This cue is *independent* of the 3D environment and needs no *a-priori* knowledge about it. It is *time-based, rotation independent* and does not need *3D reconstruction*. This cue can be extracted directly from the raw gray level data of images and does not need optical flow information, segmentation, feature tracking and pre-processing. In IQM based extraction of the VTC we can control the minimum clearance R_0 by changing the initial focus adjustments.

The results presented in this paper are for the case where the direction of motion of the observer is perpendicular to the textured surface. Currently we are working on the extension of the above approach for the cases where the direction of motion is not perpendicular to the surface. For the results presented in this paper there is no spatial smoothing on the texture or temporal smoothing on the IQM or the VTC. Adding some temporal smoothing to the IQM will significantly improve the results of the VTC. The VTC described in a previous section is good only in the region beyond the minimum clearance R_0. Currently we are investigating the nature of visual fields for points whose ranges are less than R_0. We are also working on the implementation of the VTC as a sensory feedback signal in action-perception closed-loop control system of a 6-DOF mobile robot in outdoor environments to maintain a desired safe clearance from textured surfaces.

Appendix A: The Imaging System

The amount of blur in an image is characterized by the radius (refer to Figure (A1)) of the blur circle [28-30]. The expression for the radius of the blur circle, for a camera focused to a short distance u_0, can be derived as follows [30].

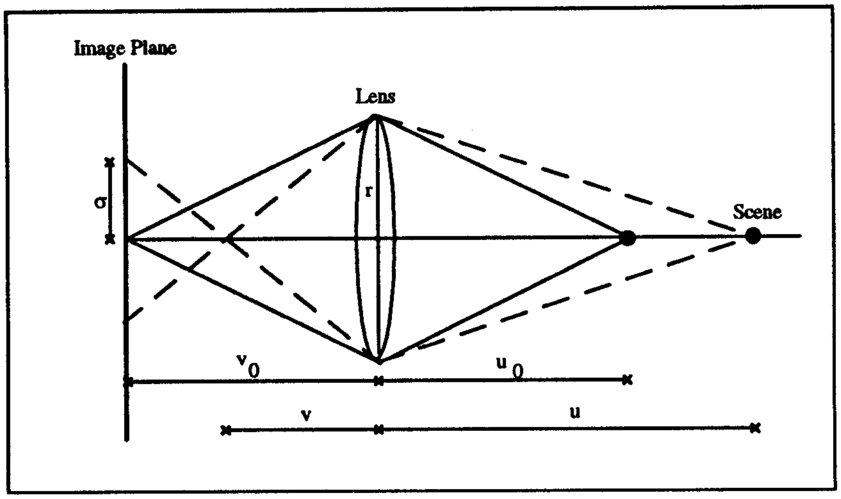


Figure A1: Imaging System

In Figure (A1) :

v_0 = distance between the image plane and the lens

u_0 = distance between the lens and the scene for which the image is in focus.

σ = radius of the blur circle (for u > v)

r = radius of the lens

F = Focal length of the lens

f = focal number of the lens

For objects in perfect focus the following Gaussian lens formula holds

$$\frac{1}{u_0}+\frac{1}{v_0}=\frac{1}{F} \tag{A1}$$

From Figure (A1) we also get the following relation

$$\frac{r}{v}=\frac{\sigma}{v_0-v} \tag{A2}$$

By combining Equation (A2) with Equation (A1) and replacing u by R we have the following relation

$$R = \frac{Fv_0}{v_0 - F - \sigma f} \qquad \text{(A3)}$$

From which the following relation is obtained:

$$\sigma = \frac{Fv_0}{f}\left(\frac{1}{R_0} - \frac{1}{R}\right) \qquad \text{(A4)}$$

where R is the distance between the object and the lens and R_0 is the distance to which the camera is focused to initially.

Appendix B: The Image Quality Measure

Let (x,y) be the spatial coordinates of an arbitrary pixel in the image, where x, y are integers and I(x,y) be the intensity at (x, y). The inter-pixel distance is denoted by δ and is defined as follows.

$$\delta = (\Delta x, \Delta y) \qquad \text{(B1)}$$

where Δx = difference between the corresponding x coordinates of two pixels; Δy = difference between the corresponding y coordinates of two pixels.

The dissimilarity between the image intensities of pixels separated by the inter-pixel distance defined in Equation (B1) is characterized by the measure *δ-dissimilarity*, which is defined as [40]:

$$\delta - dissimilarity = \{|I(x,y) - I(x+\Delta x, y+\Delta y)|\} \qquad \text{(B2)}$$

where I(x,y) = intensity at pixel (x,y); I(x+Δx, y+Δy) = intensity at pixel (x+ΔX, y+Δy).

Several other dissimilarity measures may be used instead of the one employed in Equation (B2). A detailed description of these measures is presented in [41].

Each pixel (x,y) in an image can be characterized by a matrix known dissimilarity matrix, which is basically a matrix of numbers that characterizes the dissimilarity of pixel intensities in the neighborhood of the pixel. For instance, if the texture is smooth, the dissimilarity is very low, hence the mean value of the dissimilarity matrix is low. This dissimilarity matrix is a (2Lr+1)X(2Lc+1), where Lc, Lr are positive integer constants. The (i,j) element of this matrix is the d-dissimilarity matrix as defined in Equation (B2), i = -Lc, ...,-1,0,1,...Lc and j = -Lr,...,-1,0,1,...,Lr. The (0,0) element of the dissimilarity matrix is zero since {|I(x,y)-I(x+0,y+0)}| = 0. A matrix of numbers can be generated for any pixel.

This dissimilarity matrix can be used to generate a global image variable to indicate the smoothness of texture details in an image. Next we show how to generate a global measure which indicates the texture smoothness.

We select an arbitrary window in the image plane. Let x_i and x_f be the initial and final coordinates of the window along the x-direction respectively and y_i and y_f be the initial and final coordinates of the window along the y-direction respectively. For each pixel in the window selected we compute the mean value of the dissimilarity matrix described above. Thus there are (x_f-xi)X(y_i-y_f) means for the window selected. We compute a global measure for the image smoothness by taking the mean value of the average of the dissimilarity of each pixel, i.e., mathematically it is described as follows.

$$IQM = \frac{1}{D}\sum_{x=x_i}^{x_f}\sum_{y=y_i}^{y_f}\left(\sum_{p=-L_c}^{L_c}\sum_{q=-L_r}^{L_r}\left|I(x,y)-I(x+p,y+q)\right|\right)$$

Where I(x,y) is the intensity at pixel (x,y) and x_i and x_f are the initial and final x-coordinates of the window respectively ; y_i and y_f are the initial and final y-coordinates of the window in the image respectively and Lc and Lr are positive integer constants; and D is a number defined as follows $D=(2L_c+1)\times(2L_r+1)\times(x_f-x_i)\times(y_f-y_i)$.

References

[1] C. Thorpe, M. H. Hebert, T. Kanade and S. A. Shafer (1988), "Vision and Navigation for the Carnegie Mellon Navlab", *IEEE Transactions of Pattern Analysis and Machine Intelligence*, PAMI-10, No. 3, pp. 362-373.

[2] E. D. Dickmanns, B. Mysliwetz and T. Christians (1990), "An Integrated Spatio-Temporal Approach to Automatic Visual Guidance of Autonomous Vehicles", *IEEE Transactions on Systems, Man, and Cybernetics*", Vol. 20, No. 6, pp. 1273-1284.

[3] D. Coombs and K. Roberts (1992), ""Bee-Bot":Using Optical Flow to Avoid Obstacles", *Proc. SPIE Conf. on Intelligent Robots and Computer Vision XI: Algorithms, Techniques and Active Vision*, Boston, MA., pp. 714-721.

[4] A. M. Waxman, LeMoigne and B. Srinivasan (1987), "A Visual Navigation system for Autonomous Land Vehicles", *IEEE Journal of Robotics Automation*, RA-3, No. 2, pp. 124-141.

[5] M. Turk, D. K. Morgenthaler, D. K. Gremban and M. Marra (1988), "A Vision system for Autonomous Land Vehicle Navigation", *IEEE Transactions of Pattern Analysis and Machine Intelligence*, PAMI-10, No. 3, pp. 342-360.

[6] Young Gin-Shu, Hong Tsai-Hong, M. Herman and J. C. S. Yang (1992), "Obstacle Detection and Terrain Charecterization using Optical Flow without 3D Reconstruction", Proc. *SPIE.* Vol. 1825, Intelligent Robots and Computer Vision XI, pp. 561-568.

[7] R. C. Nelson and J. Aloimonos (1989), "Obstacle Avoidance using Flow Field Divergence", IEEE *Transactions of Pattern Analysis and Machine Intelligence*, PAMI-11, No. 10., pp. 1102-1106.

[8] J. Santos-Victor, G. Sandini, F. Curotto, and Gabribaldi (1993), "Divergent Stereo for Robot Navigation: Learning from Bees", Proc. of *CVPR-93*, pp. 434-439.

[9] M. Tistarelli and G. Sandini (1990), "On Advantages of Polar and Log-Polar Mapping for Direct Estimation of Time-To-Contact from Optical Flow", Technical report *LIRA-TR3/90*, University of Genoa, Italy.

[10] D. Raviv (1992), "A Quantitative Approach to Looming", Technical Report *NISTIR 4808*, National Institute of Standards and Techology, Gaithersburg, Maryland.

[11] M. Brady and H. Wang (1992), "Vision for Mobile Robots", *Philosophical Transactions of Royal Society*, London, B337, pp. 341-350.

[12] D. N. Lee (1976), "A Theory of Visual Control of Braking Based on Information About time-to-collision", *Perception*, Vol. 5, 437-459.

[13] D. Raviv (1992), "Visual Looming", Proc. of *SPIE* Conference on Intelligent and Computer Vision XI: Algorithms, Techniques and Active Vision, Boston, MA.

[14] R. Cipolla and A. Blake (1992), "Surface Orientation and Time-to-Contact from Image Divergence and Deformation", In Proc. of *European Conference on Computer Vision*, pp. 187-202.

[15] J. R. Lishman (1981), "Vision and the Optic Flow Field", *Nature*, Vol. 293, pp. 263-264.

[16] J. J. Gibson (1979), "*The Ecological Approach to Visual Perception*", Houghton Mifflin, Boston.

[17] J. J. Gibson (1950a), "*The Perception of Visual World*", Houghton Mifflin, Boston.

[18] J. J. Gibson (1950b), "The Perception of Visual Surfaces", *American Journal of Psychology*, Vol. 63, pp. 367-384.

[19] D. N. Lee (1980), "The Optic Flow Field: The Foundation of Vision", *Philosophical Transactions of Royal Society of London*, B290, pp. 169-179.

[20] T. G. R.Bower, J. M. Broughton and M. K. Moore (1970), "The Coordination of visual and tactual input in infants", *Perception and Psychophysics*, Vol. 8, pp. 51-53.

[21] W. Schiff, J. A. Caviness and J. J. Gibson (1962), "Persistent Fear Responses in Rhesus Monkeys to the Optical Stimulus of "Looming", *Science*, Vol. 136, pp. 982-983.

[22] H. H. Nagel (1987), "On the Estimation of Optical Flow: Relations between Different Approaches and Some New results", *Artificial Intelligence*, Vol. 33, pp. 299-324.

[23] A. Singh (1991), "*Optic Flow Computation: A Unified Perspective*", IEEE Computer Society Press.

[24] J. L Barron, D. J. Fleet and S. S. Beauchemin (1992), "Performance of Optical Flow Techniques", *RPL-TR-9107*, Dept. of Computing and Information Science, Queens University, Canada.

[25] R. Bajcy and L. Liberman (1976), "Texture Gradients as Depth Cue", *Computer Graphics and Image Processing*, Vol. 5, pp. 52-67.

[26] J. Sato and R. Cipolla (1994), "Extracting the Affine Transform from Texture Moments", *TR-167*, Dept. of Engineering, Univ. of Cambridge, England.

[27] K. Joarder and D. Raviv (1994), "A New Method to Calculate Looming from Surface Texture", In Proc. of *CVPR-94*, Seattle, WA., pp. 777-780.

[28] M. Subbarao and N. Gurumoorthy (1988), "Depth Recovery from Blurred Edges", In Proc. of *CVPR-88*, pp. 498-503.

[29] S. K. Nayar (1992), "Shape from Focus System", In Proc. of *CVPR-92*, pp. 302-308.

[30] A. Pentland (1987), "A New Sense for Depth of Field", IEEE *Transactions of Pattern Analysis and Machine Intelligence*, PAMI-9, pp. 523-531.

[31] R. M. Haralick, K. Shanmugam, and I. Dinstein (1973), "Textural Features for Image Classification", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-3, No. 6, pp. 610-621.

[32] R. W. Conners and C. A. Harlow (1980), "A Theoretical Comparison of Texture Algorithms", *IEEE Transactions on Pattern Recognition and Machine Analysis*, Vol. PAMI-2, No. 3, pp. 204-222.

[33] J. S. Weszka, Dyer C. R. and A. Rosenfeld (1976), "A Comparative Study of Texture Measures for Terrain Classification", *IEEE Transactions on Systems, Man and Cybernetics*, SMC-6, pp. 269-285.

[34] M. Unser (1986), "Sum and Difference Histograms for Texture Classification", *IEEE Transactions on Pattern Recognition and Machine Analysis*, Vol. PAMI-8, pp. 118-125.

[35] W. D. Stromberg and T. G. Farr (1986), "A Fourier Based Textural Feature Extraction procedure", *IEEE Transactions on Geoscience Remote Sensing*, GE-24, pp. 722-731.

[36] M. Hassner and J. Sklansky (1980), "The use of Markov Random Fields as models of Texture", *Computer Graphics Image Processing*, Vol. 12, pp. 357-370.

[37] R. L. Kashyap (1984), "Charecterization and Estimation of two Dimensional ARMA models", *IEEE Transactions on Infromation Theory*, Vol. 30, pp. 736-745.

[38] M. M. Galloway (1975), "Texture Analysis using Grey level run lengths", *Computer Graphics Image Processing*, Vol. 4, 1975, pp. 172-179.

[39] O. R. Mitchell, C. R. Myers and W. Boyne (1977), "A Min-Max Measure for Image Texture Analysis", *IEEE Transactions on Computers*, C-26, pp. 400-414.

[40] C. M. Wu and Y. C. Chen (1992), "Statistical Feature Matrix," *Computer Vision Graphics and Image Processing: Graphical Models and Image Processing*, Vol. 54, No. 5,pp. 407-419.

[41] L. Kaufman and P. J. Rousseeuw (1990), "*Finding Groups in Data: An Introduction to Cluster Analysis*," John Wiley & Sons Inc., New York.

[42] P. Brodatz (1966), "*Textures: A Photographic Album for Artists and Designers*", Dover Publications, New York.