UNIT – IV

Structure from motion

Triangulation

Estimating the locations of 3D points from multiple images given only a sparse set of correspondences between image features.

* Problem of estimating a points 3D location which is seen from mutiple ~~cameras~~ cameras.

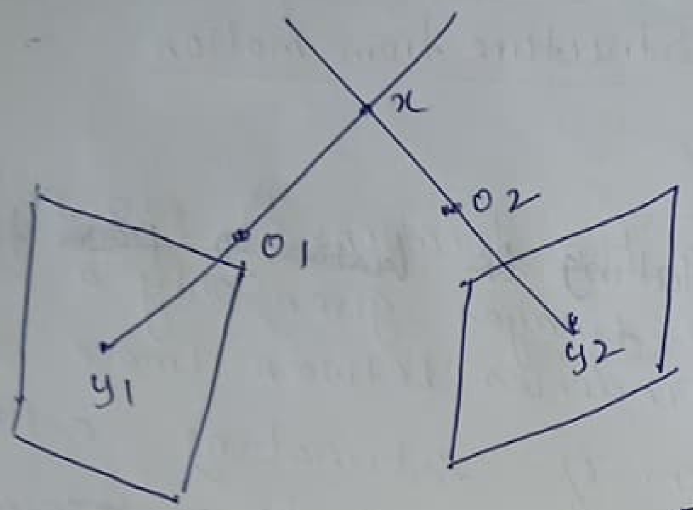
The problem of determining a points 3D position from a set of corresponding image locations and known camera positions is known as triangulation.

One of the simplest ways to solve this problem is to find the 3D point P that lies closest to all of the 3D rays corresponding to the 2D matching feature locations $\{x_j\}$ observed by cameras $\{P_j = K_j [R_j | t_j]\}$ where $t_j = -R_j C_j$ and C_j is the jth camera center.

(K_j → intrinsic Parameter; $[R_j | t_j]$ → Extrinsic Parameter of camera.)

It is necessary to know the parameters of the camera projection function from 3D to 2D for the cameras involved.

It is also referred to as reconstruction or intersection.



A 3D point x is projected onto 2 camera images through lines which intersect with each camera's focal point O_1 and O_2. The resulting image points are y_1 and y_2. If y_1 and y_2 are given and the geometry of the 2 cameras are known, the 2 projection line can be determined and it must be the case that they intersect at point x (3D point).

Properties

A triangulation can be described in terms of a function T such that

$$x \sim T(y_1', y_2', c_1, c_2)$$

where y_1', y_2' are the homogeneous co-ordinates of the detected image points and c_1, c_2 are the camera matrices.

x (3D point) is the homogeneous representation of the resulting 3D point. The ~ sign implies that T is only required to produce a vector which is equal to x upto a multiplication by a non zero scalar since homogeneous vectors are involved.

Extrinsic camera parameters

Extrinsic Parameters of a camera depend on its - Location and orientation.

Intrinsic parameters.

- focal length, field of view, resolution

Two Frame Structure from Motion

+ Simultaneous recovery of 3D structure and pose from image correspondences.

A 3D point P being viewed from 2 cameras whose relative position can be encoded by a rotation R and a translation t.

Observed location of point P in the first image $p_0 = d_0 \hat{x}_0$ is mapped into the second image by the transformation

$$d_1 \hat{x}_1 = P_1 = R P_0 + t$$
$$= R(d_0 \hat{x}_0) + t$$

Applications

1) 3D scanning (2) augmented reality (3) visual simultaneous localization and mapping.

Depends on

1) number and type of cameras used
2) whether the images are ordered.

Two frame structure from motion (SFM)

Motion

: Change of position of an object with respect to time. Eg change in rotation and translation b/w the 2 cameras.

Structure

Locations of points on the object.

SFM

Given 2 or more images (or Video frames) without knowledge of the camera poses (rotation and translations), estimate the camera poses and 3D structure of scene.

steps

1) Input: 2 images (or Video frames.
2) Detect feature points.
3) Detect feature correspondences.
4) Compute the fundamental matrix to find the pose of the second camera relative to the first camera.
5) Retrieve the relative camera 3D perspective transformation from the fundamental matrix.
6) Reconstruct corresponding 3D scene points

Feature points.

A feature is a piece of information in specific structures in the image such as points, edges or objects.

Feature point is the point which is expressive in texture. Feature detection includes methods for computing abstractions of image formation and making local decision at every image point whether there is a image feature of a given type at that point or not.

Detect feature correspondence.

To find correspondence b/w images, feature such as corner points (edges with gradient in multiple directions) are tracked from one image to the next.

· Feature detector alg is scale-invariant feature transform (SIFT).

3) Compute Fundamental matrix.

It is a relationship b/w any 2 images of the same scene that constrains where the projection of points from the scene can occur in both image.

The relation between corresponding points, which the fundamental matrix represents, is referred to as epipolar constraint matching constraint or incidence relation.

Describes the epipolor geometry of the 2 cameras.

Epipolor geometry.

* Describes the relation between the 2 resulting views when 2 cameras take a picture of the same scene from different points of view.

Projective reconstruction

It refers to the computation of the structure of a scene from images taken with uncalibrated camera, resulting in a scene structure, and camera motion that may differ from the true geometry by an unknown 3D projective transformation.

Suppose that a set of interest points / feature are identified and matched (or tracked) in several images. The configuration of the corresponding 3D points and the locations of the camera that took these images are supposed unknown. The task of reconstruction is to determine the values of these unknown quantities.

Formally, assume that a set of image points $\{x_{ij}\}$ are known, where x_{ij} represents the image co-ordinates of the jth point seen in the ith image.

It is generally not required that every point's location be known in every image, so only a subset of all possible x_{ij} are given. The SFM problem is to determine the camera projection matrices P_i and the 3D point locations X_j such that the projection of the jth point in the ith image is measured x_{ij}.

Assuming pinhole (projective) camera model, this relationship is expressed as a linear relationship

$$x_{ij} = P_i X_j$$

P_i is a 3x4 matrix of rank 3, X_j and x_{ij} are expressed in homogeneous co-ordinates, and the equality is intended to hold only upto an unknown scale factor λ_{ij}. The projection equation is

$$\lambda_{ij} x_{ij} = P_i X_j$$

In the SFM problem, cameras P_i and points X_j are to be determined, given only the point correspondence.

Projective reconstruction algorithm.

Projective reconstruction of a scene from 2 images. Suppose a set of image correspondences $x_i \leftrightarrow x_i'$; $i = 1 \dots n$ are given.

1. From the image correspondences, compute the fundamental matrix F
2. From F find the 2 camera projection matrices $P = [I|0]$ and $P' = [M|t]$
3. The corresponding 3D points X_i may be computed linearly as the least-squares solution to equations $\left.\begin{array}{l} \lambda_i x_i = P X_i \\ \lambda_i' x_i' = P' X_i \end{array}\right\}$ projection equations

This process is called triangulation.

Two-View Construction

Reconstruction problem for only 2 images. Input to the problem consists of corresponding points $x_i \leftrightarrow x_i'$; $i = 1 \dots n$ where the points x_i comes from one image and the x_i' are the corresponding points in the other.

Let the camera matrices be P and P' and let X_i be the 3D point corresponding to the image points $x_i \leftrightarrow x_i'$. The projection equations are

$$\lambda_i x_i = P X_i$$
$$\lambda_i' x_i' = P' X_i$$ where the <u>scale factors</u> λ_i and λ_i' are explicity written.

These equations may be written as a single system

$$\begin{bmatrix} P & x_i & \\ P' & & x_i' \end{bmatrix} \begin{pmatrix} X_i \\ -\lambda_i \\ -\lambda_i' \end{pmatrix} = 0$$

(ie) the determinant of the matrix (A) must be zero.

det(A) = 0 can be written as

$x_i'^T F X_i = 0$ where F is a 3x3 matrix depending only on the 2 camera matrices P and P'

The matrix F is called the fundamental matrix corresponding to the camera pair (P, P')

Self Calibration

Image calibration provides a pixel-to-real-distance conversion factor (ie the calibration factor, pixels/cm) that allows image scaling to metric units (units based on the metre, gram or second and decimal). This information can be then used throughout the analysis to convert pixel measurements performed on the image to their corresponding values in the real coord.

The purpose of camera calibration is to establish the projection from the 3D world co-ordinates to the 2D image co-ordinates. Once this projection is known, 3D information can be inferred from 2D information and vice-versa.

The camera model considered is Pinhole. The camera is assumed be perform a perfect perspective transformation. Let [su, sv, s] be the image co-ordinates where s is the non-zero scale factor.

the equation of the projection is

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = A \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} G \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = M \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

where x, y, z are world co-ordinates, A is a 3x3 transformation matrix accounting for camera sampling and optical characteristics and G is a 4x4 displacement matrix accounting for camera position and orientation. The matrix M is the Perspective transformation matrix, which relates 3D word co-ordinates and 2D image co-ordinates. The matrix G depends on six parameters called extrinsic: three defining a rotation of the camera, and three defining a translation of the camera.

The matrix A depends on a variable number of Parameters

Factorization - Projective,

- Used for recovering structure and motion from image sequences.

As we watch the video o/p from a camera moving in a three dimensional scene, we obtain an estimate of the motion of the camera as well as an idea of the geometry of the scene.

From an image sequence taken by a camera undergoing unknown motion, extract the 3D shape of the scene as well as the camera motion. This is called Structure from motion pbm. Factorization methods are used for SFM.

Factoroization algorithms depend on the mathematical possibility of decomposing a set of image measurements into the product of 2 separate factors.

image sequence $\Longleftrightarrow$ motion x shape

The projected images are considered to result from 2 factors. The relative motion between the camera and the object and the object shape.

These are composed in a bilinear form (~~bilen~~ bilinear form on a vector space V over a field K (the element ~~of~~ of which are called scalars) is a bilinear Map

$$V \times V \rightarrow K.)$$

such that if either motion or shape is constant, then the image sequence will be a linear function of the other.

The motion parameters refer to all of those parameters describing the interaction b/w the camera and the object. namely the relative orientation and translation of the object and intrinsic camera calibration parameter. These parameters may vary from image to image in the sequence, but are the same for all features in a single image.

The shape parameters describe the 3D geometric characteristics of the object and are assumed to remain constant over the sequence. The 3D co-ordinates of features on the surface of the object are used to specify shape.

The factorization method takes advantage of bilinear formulation to decompose the image measurements into a relevant motion and shape components.

The use of features is a key factor in making the factorization. It is assumed that there exists a set of features on the object that are tracked throughout the image sequence providing a complete set of feature coordinates in all images.

This assumption enables the method to focus on geometric considerations. Object shape is interpreted to mean the 3D location of the features with respect to a reference frame affixed to the object.

Object motion is the rotation and translation of this refrence frame with respect to the camera, and the image sequence means simply the co-ordinates of the projected features in the image.

The core element of factorization is the strong dependence on a bilinear formulation of structure and motion.

with appropriate choice of co-ordinates it is possible to encode both affine and perspective camera projection in the general bilinear form

↳ rays joining a point in the scene to its projection on the image plane are parallel.

$$w_{fp} = P_f M_f s_p$$

The feature co-ordinate vector w_{fp} for image f and feature p is formed as the linear sum of the product of motion parameters in matrix M_f and shape parameters in vector s_p weighted with the constant weighting or projection matrix P_f.

Constructing the equation.

we assume that there is a set of P features on an object that are projected into F images with coordinates $\{w_{fp} = (u_{fp}, v_{fp})^T \mid f = 1 \ldots F, p = 1 \ldots P\}$

Each feature has co-ordinates given by a 3×1 vector s_p for $p = 1 \ldots P$.

object motion is described by a rotation with matrix R_f and translation $t_f = (t_{fx}, t_{fy}, t_{fz})^T$ of this reference frame with respect to the camera in each image f.

A feature point p in image f will thus have position $s_p^f = R_f s_p + t_f$ with respect to the camera.

Under ortography with the z axis along the optical axis image feature w_{fp} is given by

$$w_{fp} = M_f s_p + w'_f$$ where M_f consists of the top 2 rows of the rotation matrix R_f and $w'_f = (t_{fx}\ t_{fy})^T$ is the image displacement b/w the origin of the world reference frame and the object reference frame.

If w'_f (co-ordinate system fixed to object) (no translation)

$$W^0_{fp} = M_f s_p \qquad f = 1 \ldots F, \quad p = 1 \ldots P.$$

General form

$$W = MS \qquad (1)$$

$$W = \begin{bmatrix} w^0_{11} & w^0_{12} & \cdots & w^0_{1p} \\ w^0_{21} & w^0_{22} & \cdots & w^0_{2p} \\ \vdots & & & \\ w^0_{F1} & w^0_{F2} & & w^0_{Fp} \end{bmatrix}_{2F \times P}$$

$$M = \begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_F \end{bmatrix}_{2F \times 3}$$ and $$S = [s_1\ s_2 \cdots s_p]_{3 \times P}$$

The equation W=MS contains the core of the factorization algorithm. It states that the feature locations in object centered co-ordinates can be expressed as the product of motion matrix and a shape matrix projected onto the image.

Perspective Factorization

When the object thickness is significant compared to it's depth from the camera, affine models become poor approximations to the imaging process and perspective models should be used.

The general projective camera camera model is a liner transformation of homogenous points in P^3 onto points in the plane P^2. Thus it is defined by a 3x4 Projection matrix P_p that maps object points $\bar{S} = (S^T, S_4)^T$, onto the points in the image plane $\bar{\omega} = (\omega^T, \omega_3)^T$ where these points are described in homogeneous co-ordinates

A fully automated approach to line base structure from motion is presented by werner & Zisserman. In their system, they first find lines and gp them by common vanishing points in each image. The vanishing points are then used to calibrate the camera. (ies to perform a "metric upgrade". Lines corresponding to common vanishing points are then matched using both appearance and trifocal tensor. The resulting set of 3D lines, color coded by common vanishing directions (3D orientations)

These lines are then used to infer planes and a block-structured model for the scene.

This projection occurs upto an unknown scale factor λ_{fp} called the projective depth

$$\lambda_{fp}\bar{\omega}_{fp} = P_{pf}\bar{S}_p$$

$$W = \begin{bmatrix} \lambda_{11}\bar{\omega}_{11} & \lambda_{12}\bar{\omega}_{12} & \cdots & \lambda_{1p}\bar{\omega}_{1p} \\ \lambda_{21}\bar{\omega}_{21} & \lambda_{22}\bar{\omega}_{22} & \cdots & \lambda_{2p}\bar{\omega}_{2p} \\ \vdots & & & \\ \omega_{F1}\bar{\omega}_{F1} & \lambda_{F2}\bar{\omega}_{F2} & \cdots & \lambda_{Fp}\bar{\omega}_{Fp} \end{bmatrix} = \begin{bmatrix} P1 \\ P2 \\ \vdots \\ P_F \end{bmatrix} \begin{bmatrix} \bar{S}_1 \cdots \bar{S}_p \end{bmatrix}$$

Bundle Adjustment

In photogrammetry (It is the science and technology of obtaining reliable information about physical objects and the environment through the process of recording, measuring and interpreting phographic images and patterns of electromagnetic radiant imagery and other phenomena) bundle adjustment is simultaneous refining of the 3D coordinates describing the scene geometry, the parameters of the relative motion, and the optical characteristics of the camera(s) employed to acquire the images, given a set of images depicting a no. of 3D points from different viewpoints.

It is used ~~to~~ as the last step of every feature based 3D reconstruction algorithm. It amounts to an optimization problem on the 3D structure and viewing parameters (ie, camera pose and possibly intrinsic calibration and radial distortion), to obtain a reconstruction. ~~which~~ ~~is~~

Used to minimizing the reprojection error. (The reprojection error is a geometric error corresponding to the image distance b/w a projected point and a measured one. It is used to quantify how closely an estimate of a 3D point $\hat{X}$ recreates the points true projection x.

Let P be the projection matrix of a camera and $\hat{x}$ be the image projection of $\hat{X}$ (ie) $\hat{x} = P\hat{X}$. The projection error of $\hat{X}$ is given by $d(x, \hat{x})$ where $d(x, \hat{x})$ denotes the Euclidean distance b/w the image points represented by vectors x and $\hat{x}$

Mathematical definition

Assume that n 3D points are seen in m views and let x_{ij} be the projection of the ith point on imag j.

Let V_{ij} denote the binary variables that equal 1 If point i is visible in image j and 0 otherwise. Each camera j is Parameteiized by a vector a_j and each 3D point in i by a vector b_i. Bundle adjutment minimizes the total reprojection error with respect to all 3D point and camera parameters, specically

$$\min_{a_j, b_i} \sum_{i=1}^{n} \sum_{j=1}^{m} V_{ij}\, d\left(Q(a_j, b_i), x_{ij}\right)^2$$

Where $Q(a_j, b_i)$ is the predicted projection of point i on image j and $d(x, y)$ denotes the Euclidean distance b/w the image points represented by vectors x and y.

Exploiting Sparsity.

Large bundle adjustment problems, such as those involving reconstructing 3D scenes from thousands of Internet photographs can require solving non-linear least squares problems with millions of measurements (feature matches) and tens of thousands of unknown parameters (3D point positions and camera pose)

Unless some care is taken, these kinds of pbm can become intractable since the direct solution of dense least squares problems is cubic on the no. of unknowns.

Fortunately, structure from motion is a bipartite pbm in structure and motion. Each feature point x_{ij} in a given image depends on one 3D point position p_i and one 3D camera pose (R_j, c_j).

Sparse cholesky factorization technique is used for the solution of bundle adjustment problems.

Cholesky decomposition or factorization is a powerful numerical optimization technique that is widely used in linear algebra. It decomposes an Hermitian, positive definite matrix into a lower triangular and its conjugate component.

Thor
table

Constrained structure and motion

The most general algorithms for structure from motion make no prior assumptions about the objects or scenes that they are reconstructing. In many cases, however, the scene contains higher level geometric primitives such as lines and planes. These can provide information complementary to interest points and also serve as useful building blocks for 3D modeling and visualization.

Sometimes, instead of exploiting regularity in the scene structure, It is possible to take advantage of a constrained motion model.

For eg, if the object of interest is rotating on a turntable (roundtable) (ie) around a fixed but unknown axis, specialized techniques can be used to recove the motion.

Line based Technique.

schmid describe a widely used technique for matching 2D lines based on the average of 15 x 15 pixel correlation scores evaluated at all pixels along their common line segment intersection.

In their system, the epipolor geometry is assumed to be known. eg computed from point matches.

For wide baselines, all possible homographies corresponding to planes passing through the 3D line are used to wrap pixels and the maximum correlation score is used.

For triplets of images, the trifocal tensor is used to verify that the lines are in geometric correspondance before evaluating the correlations b/w line segment.

Hierarchical structure and motion

To estimate the motion between 2 or more images, a suitable error metric must first be chosen to compare the images. Once this has been established, a suitable search technique must be derived. The simplest technique is to exhaustively try all possible alignments. (ie) to do a full search. Each block of pixels in one frame is compared to every possible block in the next frame to find the best match, and the corresponding motion vector.

Motion estimation

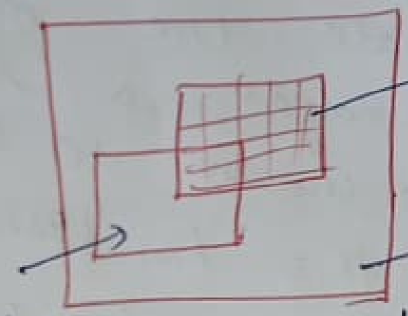
Motion estimation is the process of determining motion vectors that describe the transformation from one 2D image to another.

Successive video frames may contain the same objects (still or moving). Motion estimation examines the movement of objects in an image sequence to try to obtain vectors representing the estimated motion.

Motion vector

A motion vector is calculated by finding a correspondence between rectangles at time t, and rectangles at time t-1, where t is the frame index in a Video.

$$\vec{V} = \arg\min || I(x,y,t) - I(x-v_1, y-v_2, t-1)$$



In real video scenes, motion can be combination of translation and rotation. motion estimation algorithm make the following assumptions.

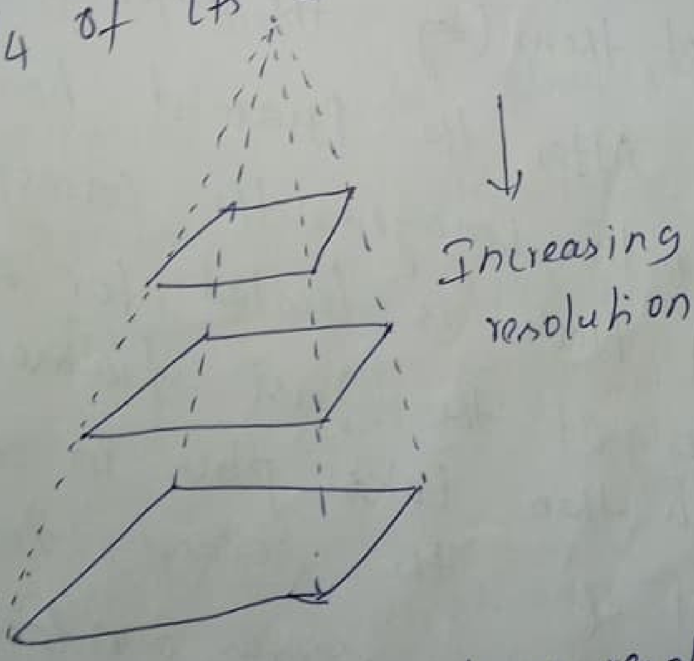
1. Objects move in translation in a plane that is parallel to the camera plane. ie the effects of camera zoom, and object rotations are not considered.

2. Illumination is spatially and temporally uniform.

3. Occlusion of one object by another, and uncovered background are neglected.

→ occlusion in an image occurs when an object hides a part of another object.

Hierarchical estimation of the motion vector field also known as ar pyramid search is widely used for motion estimation. In hierarchical estimation, both frames undergo a process of size and resolution reduction. several levels are constructed, each containing the same image as the previous level, having both dimensions reduced by a certain factor (usually 2).
The result is a pyramid, where the lowest level is the initial image, and each level above it is the same image at 1/4 of its size.



In order to create a lower resolution image from the initial one, 2 approaches can be used.
1. Mean intensity 2) subsampling.

In the case of grey-level image, for the mean intensity approach, each block of 4 pixels is replaced by one, having their mean intensity. ie

$$g_L(p,q) = \left[\frac{1}{4}\sum_{u=0}^{1}\sum_{v=0}^{1} g_{L-1}(2p+u, 2q+v)\right]$$

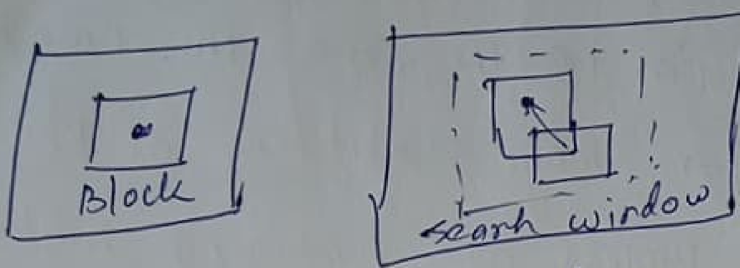
$$1 \leq L \leq 2.$$

where $g_L(p,q)$ is the pixel intensity of pixel (p,q) of the Lth level, and $g_0(p,q)$ denotes pixel (p,q) of the original image.

subsampling is another approach for reducing an images size. During subsampling each block of pixels is replaced by only one of them (eg the upper leftmost)

After the pyramid has been created for both images, the corresponding higher level block is located, for each 0-level block of the first frame. A full search then takes place in the higher level of the second frame. This means that a search window is defined in the second frame, and for each block in the first frame all candidate motion vectors are evaluated.

This is achieved by comparing all the blocks in the search window to the block in the first frame whose vector is sought



Frame k Frame k+1

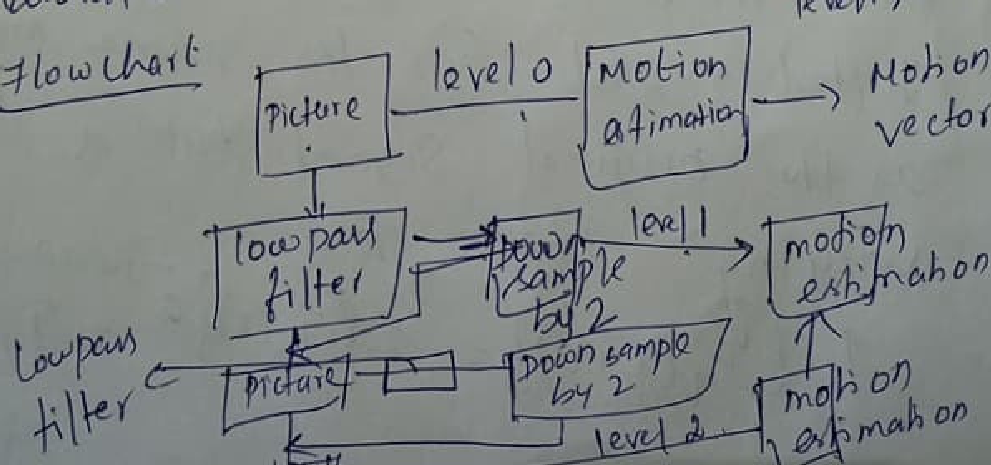
In order to compare blocks, a measure of the block difference has to be established. The most widely used block distance measure is the Mean absolute Difference.

$$MAD(i,j) = \frac{1}{mn} \sum_{k} \sum_{l} \left| g_f(k,l) - g_{f-1}(k+i, l+j) \right|$$

where ~~m,n~~ (m,n) are the block dimensions $g_f(k,l)$ signifies the intensity of pixel (k,l) in frame f and (i,j) is the candidate motion vector.

Parameters to be specified. (block size, no. of levels, scaling factor)

Flowchart



Fourier-based alignment

When the search range corresponds to a significant fraction of the larger image (as is the case in image stithing → Image stitching or photo stitching is the process of combining multiple photographic images with overlapping fields of view to produce a segmented Panorama or high resolution image), the hierarchical approach may not work well, since it is often not possible to coarsen (make) the representation too much before significant features are blurred away. In this case, a Fourier based approach may be preferable.

any wide angle view or representation of a physical space

Fourier-based alignment relies on the fact that the Fourier transform of a shifted signal has the same magnitude as the original signal but a linearly varying phase ie.

$$F\{I_1(x+u)\} = F\{I_1(x)\}e^{-ju\cdot\omega} = I_1(\omega)e^{-ju\cdot\omega}$$

where ω is the vector-valued angular frequency of the Fourier transform and we use caligraphic notation $\mathcal{I}_1(\omega) = \mathcal{F}\{I_1(x)\}$ to denote the Fourier transform of a signal.

Another useful property of Fourier transform is that convolution in the spatial domain corresponds to multiplication in the Fourier domain. Thus the Fourier transform of the cross-correlation function E_{cc} can be written as

$$\mathcal{F}\{E_{cc}(u)\} = \mathcal{F}\{\sum_i I_0(x_i) I_1(x_i+u)\} = \mathcal{F}\{I_0(u) \bar{*} I_1(u)\} = \mathcal{I}_0(\omega)\hat{\mathcal{I}}_i(\omega)$$

where $f(u) \bar{*} g(u) = \sum_i f(x_i) g(x_i+u)$ is the correlation function. (ie) the convolution of one signal with the reverse of the other. and $\mathcal{I}_1^*(\omega)$ is the complex conjugate of $\mathcal{I}_1(\omega)$. This is because convolution is defined as the summation of one signal with the reverse of the other.

Thus to efficiently evaluate ECC over range of all possible values of u, we take the Fourier transforms of both images $I_0(x)$ and $I_1(x)$, multiply both transform together (after conjugating the second one) and take the inverse transform of the result.

While Fourier-based convolution is often used to accelerate the computation image correlation, it can also be used to accelerate the sum of squares difference function (and its variants.

Consider the SSD formula. Its Fourier transform can be written as

$$F\{E_{SSD}(u)\} = F\left\{ \sum_i [I_1(x_i+u) - I_0(x_i)]^2 \right\}$$

$$= \delta(\omega) \sum_i [I_0^2(x_i) + I_1^2(x_i)] - 2 I_0(\omega) I_1^*(\omega).$$

Thus the SSD function can be computed by taking twice the correlation function and subtracting it from the sum of the energies in the 2 images.

Incremental refinemente.

In general Image stabilization and stitching applications require much higher accuracies to obtain acceptable results.

To obtain better sub-pixel estimate we can use the alg which evaluate several discrete (integer or fractional) values of (u,v) around the best value found so far and to interpolate the matching score to find an alalytic minimum.

A more commonly used approach is to perform gradient descent on the SSD energy function. Using a Taylor series expansion of the image function.

$$E_{LK-SSD}(u+\Delta u) = \sum [I_1(x_i + u + \Delta u) - I_0(x_i)]^2$$

$$\approx \sum_i [I_1(x_i + u) + J_1(x_i + u)\Delta u - I_0(x_i)]^2$$

$$= \sum_i [J_1(x_i + u)\Delta u + e_i]^2$$

where

$J_1(x_i+u) = \nabla I_1(x_i+u) = \left(\frac{\partial I_1}{\partial x}, \frac{\partial I_1}{\partial y}\right)(x_i+u)$

is the image gradient or Jocobian at (x_i+u) and

$e_i = I_1(x_i+u) - I_0(x_i)$ is the current intensity error.

The gradient at a particular sub-pixel location (x_i+u) can be computed using a variety of techniques, the simplest of which is to simply take the horizontal and vertical differences b/w pixels x and $x+(1,0)$ or $x+(0,1)$.

The linearized form of the incremental update to the SSD error is often called the optical flow constraint or brightness constancy constraint equation

$I_x u + I_y v + I_t = 0$. where the subscripts in I_x and I_y denote spatial derivaties, and I_t is called the temporal derivative which makes sense if we are computing instantaneous velocity in a video sequence.