Direct Monocular Articulated Visual Odometry

Direct Monocular Visual Odometry

We want to solve this problem: given a current camera image, previous camera image a known camera intrinsic matrix, and a guess of the previous depth image, derive the most likely camera motion between the current and previous images (Fig. 1). Direct monocular visual odometry solves this by reprojecting all the points from the previous camera frame into the current camera frame, and comparing the resulting pixel intensities. That is, we want to minimize:

$$E(\xi) = \sum_i \left[I_{t-1}(p_i) - I_t(\omega(p_i, D_{t-1}(p_i), \xi) \right]^2 \tag{1}$$

$$= \sum_i r_i(\xi) \tag{2}$$

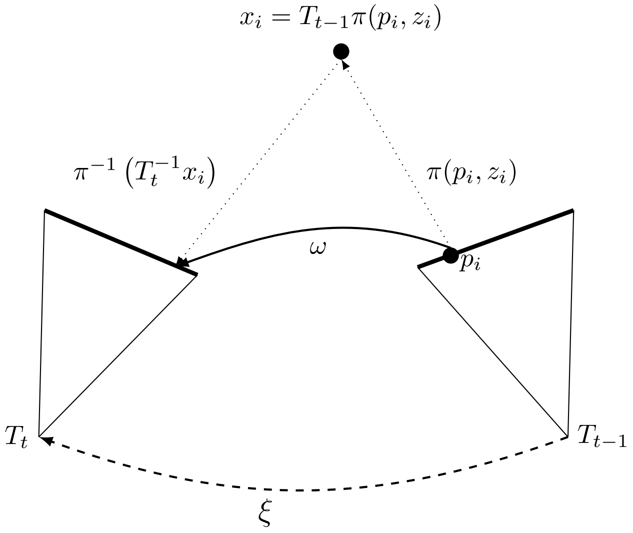
where $p_i \in \mathbb{R}^2$ is an image point, I_t, I_{t-1} are the current and previous camera images, D_{t-1} is the previous depth image, ξ is an incremental motion in $SE(3)$, and ω is the warping function that projects a pixel from the previous image onto the current image. Specifically, ω is of the form

$$\omega(p_i, z_i, \xi) = \pi^{-1} \left((\xi \circ T_{t-1}) \pi(p_i, z) \right) \tag{3}$$

where $\pi : \mathbb{R}^2 \to \mathbb{R}^3$ is the pinhole camera projection function of the camera, $\circ$ is a compositional operator that applies a small motion to the previous camera pose T_{t-1}.

Equation 2 is generally minimized using Gauss Newton, with the update:

$$\Delta\xi = (\mathbf{J}^T\mathbf{J})^{-1} \sum_i (\mathbf{J}_i r_i(\xi)) \tag{4}$$



$x_i = T_{t-1}\pi(p_i, z_i)$
$\pi^{-1}\left(T_t^{-1}x_i\right)$
$\pi(p_i, z_i)$
ω
p_i
T_t
T_{t-1}
ξ

Figure 1: An image point p_i is unprojected out of camera frame T_{t-1}, and then projected back onto the image of the camera at T_t. The offset in $SE(3)$ between the camera frames is ξ. The warp between the image point in frame $t-1$ and frame t is given by the function ω.

Now, the Gauss-Newton update contains the term $\mathbf{J}$, which is the partial derivative of the stacked residuals with respect to a small motion ξ. It has the form:

$$\mathbf{J} = \left[\mathbf{J}_1 | \dots | \mathbf{J}_K\right] \tag{5}$$

where each $\mathbf{J}_i$ is the jacobian of a particular residual with respect to a small motion ξ, and has the form:

$$\mathbf{J}_i = \nabla I\big|_{\omega(x_i, T_t)} \frac{\partial \pi}{\partial p_i}\big|_{T_t p_i} \frac{\partial T p_i}{\partial T}\big|_{T_t} \frac{\partial T T_t}{\partial T} \frac{\partial \exp(\hat{\xi})}{\partial \xi}\big|_0 \tag{6}$$

Equation is quite complicated. It has 4 terms to convert between image gradients to motion on the Lie manifold of $SE(3)$:

- $\nabla I\big|_{\omega(x_i, T_t)} \in \mathbb{R}^{1\times 2}$ is the gradient of the image at time t evaluated at the projected point. This is easily computed by finite-differencing the image.
- $\frac{\partial \pi}{\partial p_i}\big|_{T_t p_i}$ is the derivative of the camera's projection function with respect to a change in the projected point from the previous image. This can be computed in closed form.
- $\frac{\partial T p_i}{\partial T}\big|_{T_t} \in \mathbb{R}^{3\times 12}$ is the partial derivative of the projected image point with respect to the current transformation. This can be directly computed.

- $\frac{\partial T T_t}{\partial T}\frac{\partial \exp(\hat{\xi})}{\partial \xi}\big|_0 \in \mathbb{R}^{12\times 6}$ converts a change in pose into an infinitesimal increment on the Lie manifold. This is known in closed form, and relies on the "exponential map" of $SE(3)$

note that

$$\frac{\partial \pi}{\partial p_i}\Big|_{T_t p_i} = \begin{pmatrix} f_x \frac{1}{z'} & 0 & -f_x \frac{x'}{z'^2} \\ 0 & f_y \frac{1}{z'} & -f_y \frac{y'}{z'^2} \end{pmatrix} \tag{7}$$

where $T_t p_i = (x', y', z')$, and f_x, f_y come from the camera intrinsics.

Optimization then proceeds by the standard Gauss-Newton rules, with necessary conversions between poses in $T \in SE(3)$, and increments ξ along the 6DOF Lie manifold.

This is done (in contrast to using something like a quaternion or rotation matrix representation for ξ) since it prevents degenerate solutions and allows Gauss-Newton to proceed with exactly 6 variables.

The Articulated Case

When we don't have a free-floating camera, and instead have a camera on the end of a constrained, articulated manipulator (Fig. 2), the problem is slightly different. A robot manipulator with configuration $q \in \mathbf{R}^N$ has a forward kinematics function $F(q) : \mathbf{R}^N \rightarrow SE(3)$ defining the pose of the sensor. We can also write $F(q, x) : \mathbf{R}^N \times \mathbf{R}^3 \rightarrow \mathbf{R}^3$ which transforms a point in the camera frame into the world frame given a configuration of the robot.

In this case, the cost function we'd want to minimize is:

$$E(\Delta q) = \sum_i \left[I_{t-1}(p_i) - I_t(\omega(p_i, D_{t-1}(p_i), \Delta q) \right]^2 \tag{8}$$

$$= \sum_i r_i(\xi) \tag{9}$$

where the warping function is now:

$$\omega(p_i, z_i, \Delta q) = \pi^{-1}\left(F(q_{t-1} + \Delta q, \pi(p_i, z)\right) \tag{10}$$

Notice that the $SE(3)$ compositional operator ($\circ$) has been replaced by simple addition. This is important, because it has removed the complicated mess of converting between Lie manifold coordinates and transformation matrices. Instead, we can do the whole optimization in the configuration space of the robot, which is Euclidean. That means the Jacobian is now:

$$\mathbf{J}_i = \nabla I\big|_{\omega(p_i, z_i, 0)} \frac{\partial \pi}{\partial p_i}\Big|_{x_i} \frac{\partial F(q, x_i)}{\partial q}\Big|_{q_{t-1}} \tag{11}$$

and it has size $N \times 1$, rather than 6×1. The complicated Lie manifold conversions get replaced by the simple translational manipulator Jacobian $\frac{\partial F(q, x_i)}{\partial q}\big|_{q_{t-1}}$,

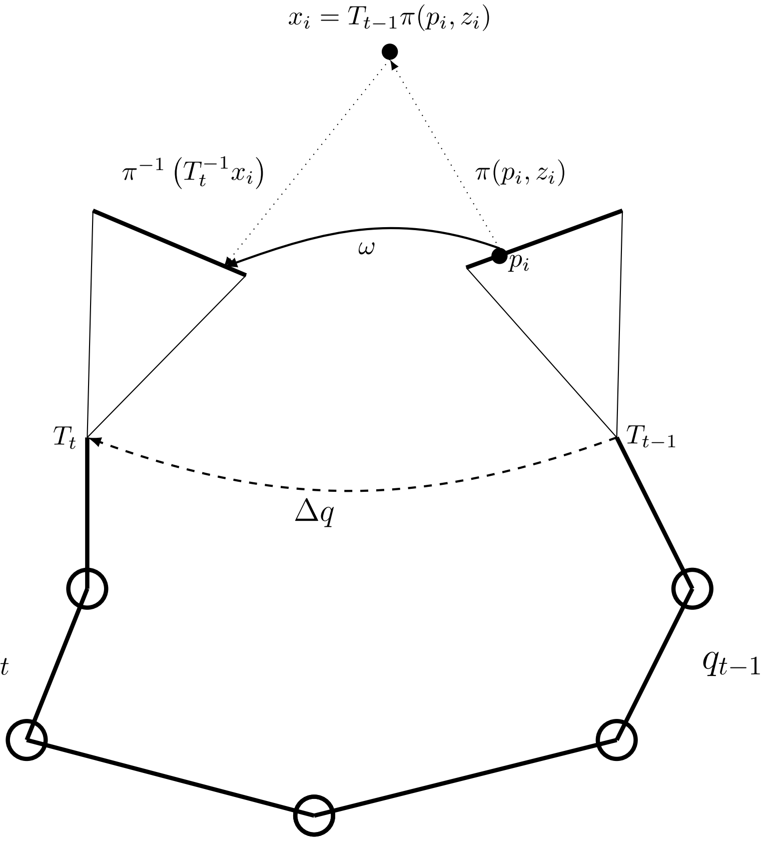


Figure 2: A camera mounted to a moving robot arm. The case is the same as in Fig. 1, except the transformation between the two camera frames is given by Δq, instead of ξ, since the camera motion is constrained by the kinematics of the robot.

evaluated for a point projected from the previous image, which can be computed easily in closed form.