experimental comparison of some of these algorithms on image retrieval datasets. You can also find more details on related techniques and systems in Section 6.2.3 on visual similarity search, which discusses global descriptors that represent an image with a single vector (Arandjelovic, Gronat et al. 2016; Radenović, Tolias, and Chum 2019; Yang, Kien Nguyen et al. 2019; Cao, Araujo, and Sim 2020; Ng, Balntas et al. 2020; Tolias, Jenicek, and Chum 2020) as alternatives to bags of local features, Section 11.2.3 on location recognition, and Section 11.4.6 on large-scale 3D reconstruction from community (internet) photos.

7.1.5 Feature tracking

An alternative to independently finding features in all candidate images and then matching them is to find a set of likely feature locations in a first image and to then search for their corresponding locations in subsequent images. This kind of detect then track approach is more widely used for video tracking applications, where the expected amount of motion and appearance deformation between adjacent frames is expected to be small.

The process of selecting good features to track is closely related to selecting good features for more general recognition applications. In practice, regions containing high gradients in both directions, i.e., which have high eigenvalues in the auto-correlation matrix (7.8), provide stable locations at which to find correspondences (Shi and Tomasi 1994).

In subsequent frames, searching for locations where the corresponding patch has low squared difference (7.1) often works well enough. However, if the images are undergoing brightness change, explicitly compensating for such variations (9.9) or using normalized cross-correlation (9.11) may be preferable. If the search range is large, it is also often more efficient to use a hierarchical search strategy, which uses matches in lower-resolution images to provide better initial guesses and hence speed up the search (Section 9.1.1). Alternatives to this strategy involve learning what the appearance of the patch being tracked should be and then searching for it in the vicinity of its predicted position (Avidan 2001; Jurie and Dhome 2002; Williams, Blake, and Cipolla 2003). These topics are all covered in more detail in Section 9.1.3.

If features are being tracked over longer image sequences, their appearance can undergo larger changes. You then have to decide whether to continue matching against the originally detected patch (feature) or to re-sample each subsequent frame at the matching location. The former strategy is prone to failure, as the original patch can undergo appearance changes such as foreshortening. The latter runs the risk of the feature drifting from its original location to some other location in the image (Shi and Tomasi 1994). (Mathematically, small misregistration errors compound to create a Markov random walk, which leads to larger drift over time.)

Jenicek, and Chum 2020) as alternatives to bags of local features, Section 11.2.3 on location recognition, and Section 11.4.6 on large-scale 3D reconstruction from community (internet) photos.

7.1.5 Feature tracking

An alternative to independently finding features in all candidate images and then matching them is to find a set of likely feature locations in a first image and to then search for their corresponding locations in subsequent images. This kind of detect then track approach is more widely used for video tracking applications, where the expected amount of motion and appearance deformation between adjacent frames is expected to be small.

The process of selecting good features to track is closely related to selecting good features for more general recognition applications. In practice, regions containing high gradients in both directions, i.e., which have high eigenvalues in the auto-correlation matrix (7.8), provide stable locations at which to find correspondences (Shi and Tomasi 1994).

In subsequent frames, searching for locations where the corresponding patch has low squared difference (7.1) often works well enough. However, if the images are undergoing brightness change, explicitly compensating for such variations (9.9) or using *normalized* cross-correlation (9.11) may be preferable. If the search range is large, it is also often more efficient to use a hierarchical search strategy, which uses matches in lower-resolution images to provide better initial guesses and hence speed up the search (Section 9.1.1). Alternatives to this strategy involve learning what the appearance of the patch being tracked should be and then searching for it in the vicinity of its predicted position (Avidan 2001; Jurie and Dhome 2002; Williams, Blake, and Cipolla 2003). These topics are all covered in more detail in Section 9.1.3.

If features are being tracked over longer image sequences, their appearance can undergo larger changes. You then have to decide whether to continue matching against the originally detected patch (feature) or to re-sample each subsequent frame at the matching location. The former strategy is prone to failure, as the original patch can undergo appearance changes such as foreshortening. The latter runs the risk of the feature drifting from its original location to some other location in the image (Shi and Tomasi 1994). (Mathematically, small misregistration errors compound to create a *Markov random walk*, which leads to larger drift over time.)

A preferable solution is to compare the original patch to later image locations using an affine motion model (Section 9.2). Shi and Tomasi (1994) first compare patches in neighboring frames using a translational model and then use the location estimates produced by

IFT + FLANN | Lowe ratio = 0.7 | NOTE: features were NOT limited to 10; MIN_MATCH_COUNT = 10

Computer Vision: Algorithms and Applications, 2nd ed. (final draft, Sept. 2021)

experimental comparison of some of these algorithms on image retrieval datasets. You can also find more details on related techniques and systems in Section 6.2.3 on visual similarity search, which discusses global descriptors that represent an image with a single vector (Arandjelovic, Gronat *et al.* 2016; Radenović, Tolias, and Chum 2019; Yang, Kien Nguyen *et al.* 2019; Cao, Araujo, and Sim 2020; Ng, Balntas *et al.* 2020; Tolias, Jenicek, and Chum 2020) as alternatives to bags of local features, Section 11.2.3 on location recognition, and Section 11.4.6 on large-scale 3D reconstruction from community (internet) photos.

7.1.5 Feature tracking

An alternative to independently finding features in all candidate images and then matching them is to find a set of likely feature locations in a first image and to then search for their corresponding locations in subsequent images. This kind of detect then track approach is more widely used for video tracking applications, where the expected amount of motion and appearance deformation between adjacent frames is expected to be small.

The process of selecting good features to track is closely related to selecting good features for more general recognition applications. In tractice, regions containing high gradients in both directions, i.e., which have high eigenvalues in the auto-correlation matrix (7.8), provide stable locations at which to find correspondences (Shi and terms 1994).

In subsequent frames, searching for locations where the corresponding patch has low squared difference (7.1) often works well enough. However, if the automate are undertooning brightness change, explicitly compensating for such surfations (9) or using a treative of cross-correlation (9.11) may be preferable. If the search range is large, if it also often more efficient to use a hierarchical search strategy, which uses matches in lower resolution at masses to provide better initial oursess and hence speed up the search (Section 9.1). Itematical to this strategy involve learning what the appearance of the patch being tracked should be and then searching for it in the vacinity of us predicted position (Avidan 2001, Ison, and Thomas 2002; Williams, Blake, and Cipolia 2003). These topics are all covered in more detail in Section 9.1.3.

If features are being tracked over longer image sequences, their appearance can undergo larger changes. You then have to decide whether to continue man hing against the originally detected patch (feature) or to re-sample each subsequent drame at the matching location. The former strategy is prone to failure, as the original patch or undergo appearance charges such as foreshortening. The latter runs the risk of the feature drafting from its original location to some other location in the image (Shound Tongai 1991). (Mathematically, small menegistration errors compound to create a Markor random risk which leads to have draft over time.)

The sale of the state of the same and the sa To make econor of the containing high practice was containing high practices in the directions he which have high eigenvalues in the auto correlation matrix (7.8), provide deegtes for jocations where the corresponding patch has low undergo the realizable where the corresponding however if the images are undergo to jocations where the corresponding patch has low undergo the course of the images are undergo the images are the discharge supported in the state of the ourselation of the many be prest rable. If the search range is large, it is also often more Mich uses matches in lower resolution images Alternatives and hence speed up the search (Section 9.1.1). Alternatives to this trategy Awalve learning what the appearance of the Patch being tracked should be and head seems that the appearance of the Patch being tracked should be and head seems that the appearance of the Patch being tracked should be and head seems that the appearance of the Patch being tracked should be and head seems that the appearance of the Patch being tracked should be and head seems that the appearance of the Patch being tracked should be and head seems to be a seem to be a s then searching for it in the visiting of its Aredicted Position (Avidan 2001; Jurie and Dhome These topics are all covered in more detail in If features are being tracked over longer image sequences, their appearance can undergo larger changes. You then have to decide whether to continue matching against the originally detected patch (feature) or to re-sample each subsequent frame at the matching location. The former strategy is prone to failure, as the original parch can undergo appearance changes such as foreshortening. The latter runs the risk of the feature drifting from its original location to 2002: Williams, Blakes, and Cipcula 2003). Tu-Some other location in the image (Shrand Tomas) 1994). (Mathematically, small misreeise tration errors compound to create a Markov random walk, which leads to larger drift over A Preferable solution is to compare the original patch to later image locations using an affine motion model (Section 9.2). Shi and Tomasi (1994) first compare patches in neight boring frames using a translational model and then use the location estimates produced by this step to initialize an affine registration between the patch in the current frame and the