

Object flow

Anonymous ACCV 2014 submission

Paper ID ***

Abstract. Motion analysis in image sequences has undoubtedly shown good progress in terms of its two main research branches. Optical flow estimation and object visual tracking have been mostly studied as isolated problems, and high accuracy algorithms are available when needed as independent bricks. This paper presents a framework for combining object tracking techniques with optical flow methods aiming towards a precise motion description for objects in video sequences. Firstly, we introduce a method to extend max-flow min-cut based segmentation techniques to videos, without adding the computational load of performing a graph cut optimization approach for 3-dimensional graphs. This is done by exploiting the inherent foreground-background separation hints given by object trackers, and the novel concept of superpixel flow. Then, we show that long-motion awareness obtained from object tracking, together with a per frame object segmentation can improve the precision of the object motion description in comparison to several optical flow techniques. We may call the proposed approach Object flow as it offers a dense and semantic aware description of the current motion state of the studied object.

1 Introduction

Object tracking and optical flow are two of the main components in the computer vision toolbox, and have been focus of great research efforts, leading to significant progress in the last years [16][17]. The object tracking problem in videos consists on estimating the position of a target in every frame, given an initial position. On the other hand, the optical flow between a pair of frames consists on finding a displacement vector for each pixel of the first image, namely a *dense motion or displacement field*. Even though for several applications a complete (i.e. for every pixel) motion-field is needed, other applications like human-computer interaction, object editing in video or structure-from-motion, may only focus on an interest object and thus, only a subset of motion vectors is required. In such scenarios combining optical flow and object tracking in a unified framework appears useful and the precision of the object motion description could be enhanced. For instance, even with modern optical flow approaches, the long term dense motion estimation remains a challenge [20][22]. At large, object trackers provide a more robust, longer term motion estimation featuring a global description of an object, specially after recent works based on tracking-by-detection approaches [16][23][24]. On the other side, they lack the (sub) pixel precision of dense optical flow estimators, as well as a deeper use of contextual information for bundle

045 motion vector estimation. Even more, object trackers and optical flow give pre-
 046 cious hints for other fundamental tools such as object segmentation in video.
 047 Nevertheless, these two techniques were not deeply studied in the literature as a
 048 unified problem. Though optical flow has been widely used as a motion feature
 049 for object tracking [25], feeding a dense motion estimator with tracking informa-
 050 tion is not being fully exploited. This being said, we introduce a new problem
 051 which we call object flow. Thus, for a given object of interest, the object flow
 052 is the set of displacement vectors for every pixel that belong to the target in
 053 a first frame, towards another frame of the sequence. In other words, a dense
 054 displacement field constrained to the spatial support of the object. Note that by
 055 definition this induces a segmentation of the target and of the motion field.

056 We can define more precisely the object flow by starting with an image
 057 sequence, say $I_t, t : 0..N - 1$, and an initial position of the interest object in
 058 the first frame of this sequence. Let $\mathcal{R} \in \Omega$ be the region corresponding to the
 059 support of the object in the bi-dimensional grid Ω . Then, the object flow, $\mathcal{O}(x)$,
 060 is defined as $\mathcal{O}(x) = d_{0,t}(x), \forall x \in \mathcal{R}$.

061 A straightforward solution to this problem would be to compute the optical
 062 flow field between a pair of frames, and to apply a segmentation mask to recover
 063 the desired motion vectors. Nevertheless, this approach carries several problems.
 064 For example, a globally computed optical flow method can affect small objects
 065 motion, because of the common use of heavy regularization prior. Moreover,
 066 finding the pixels that belong to the interest object in several frames is a difficult
 067 problem. We propose an approach to reduce these problems.

069 2 The object flow estimation pipeline

070
 071 Following the definition of the problem given above we have devised a system for
 072 computing the object flow. The Fig. 1 shows a simplified block diagram of the
 073 proposed system. It is important to recognize the two main components of our
 074 work-flow: object tracking and segmentation, and pixel-wise flow computation
 075 on the pixels of interest. The scheme is completed by feeding back the flow
 076 information to improve the tracker as depicted in the figure. For instance, one
 077 can make the most of dense displacement vectors to refine the motion of the
 078 target, and the segmentation information can be used to improve the sampling
 079 process of the learning stage in several trackers by detection methods [22]. Thus,
 080 the tracker and motion flow algorithm can work for mutual enhancement.

081 **Tracking.** The object tracking in the object flow pipeline can be selected ac-
 082 cording to a specific need for a given application. Here we prefer recent methods
 083 based on tracking-by-detection given that they are in the top of modern bench-
 084 marks for object tracking [16] in terms of accuracy and stability to initialization
 085 changes. Even more, as previously mentioned tracking-by-detection techniques
 086 can benefit from the background-foreground separation of the next stage in the
 087 object flow pipeline. For instance, the sampling process in the *Struck* tracker can
 088 be improved by selecting positive samples only inside the region of the target
 089 object.

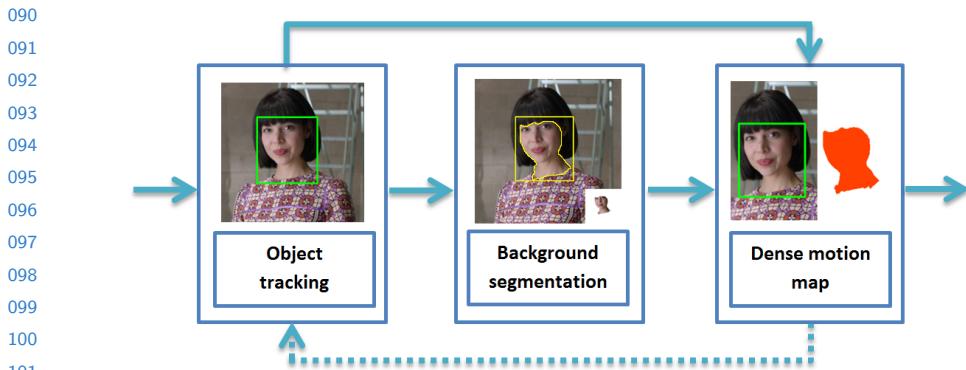


Fig. 1. Block diagram of the object flow pipeline.

Object Support Extraction. Determining the spatial support of the object in a given frame benefits from the output of the tracker. Appearance cues alone, learned inside and outside the tracking window can result in a misleading modeling of the foreground and the background. In contrast, we propose to perform foreground-background segmentation by tracking background pixels surrounding the target, thanks to the tracker output. Thus, the pixels that are initially outside the tracker window, are followed through the sequence and as long as they enter the tracked region, they can be safely labeled as background. This idea can be observed in the Fig. 14, where the object window given by the tracker (green) loosely separates the foreground from the background. Points outside the tracker (blue) are labeled as background in previous frames, and as they enter the tracker window (red points), they can be used to improve the modeling of the foreground and the background. The red window is used to save computational power by avoiding to track points that are too far from the interest object.

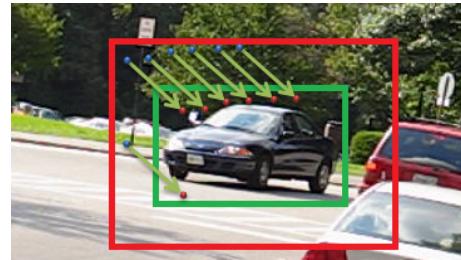
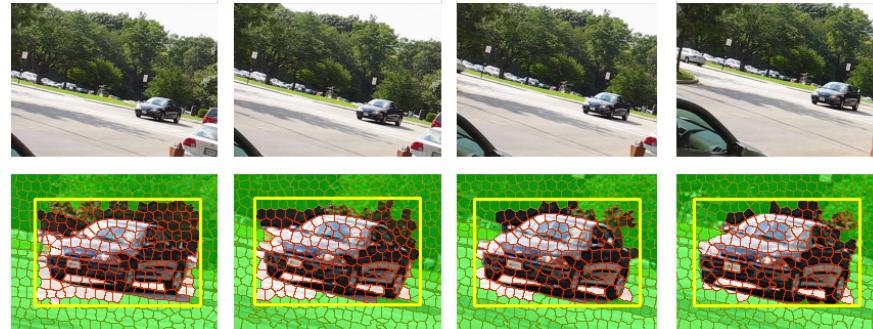


Fig. 2. Example image of points entering a tracking region (green) due to object motion in a video sequence.

135 Tracking pixels as independent points, however, carries several problems.
 136 First, to track all the background points means to compute the optical flow
 137 between the image pair, which can become very inefficient when taking into
 138 account large objects. Second, the point tracking techniques causes some drift in
 139 the trajectories, which can lead to wrong labelling as background. Finally, the
 140 background labelling could be more dense, and less complex if superpixels are
 141 tracked instead of pixels. After several frames, the labeled superpixels will almost
 142 completely cover the unwanted areas in a dinamyc scene. We call this process
 143 background segments tracking. Fig. 15 shows this idea in a real scenario. From
 144 left to right, initially the superpixels with elements outside the bounding box are
 145 labeled as background (green), then, as the sequence runs, the labeled superpixels
 146 flow inside the window, propagating the background mask. The segmentation is
 147 then refined applying the grab-cut method ([18][15]) guided by the background
 148 labeling, which works a an initialization, replacing the usual user interaction.



150
 151
 152
 153
 154
 155
 156
 157
 158
 159
 160
Fig. 3. Background segments automatic labeling and propagation, the flow goes from
 161 left to right. The superpixels that are outside the tracker window (yellow) are labelled
 162 as background (green) in the first frame. These labels are propagated when the sequence
 163 runs.
 164

165 In order to perform the background segments tracking, we propose a super-
 166 pixel matching technique, which we call superpixel flow. The objective of the
 167 superpixel flow is to find the best match for every superpixel p in the first frame
 168 with one (p') in the next frame, while holding a global flow-like behaviour.

169 Thus, the superpixelization should maintain certain size homogeneity within
 170 a single frame. Some super pixel techniques can cope with this requirement
 171 [9][10]. For the experiments presented in this work, we prefer the SLIC method
 172 [9], which gives good results in terms of size homogeneity and compactness of
 173 the superpixelization.

174 Inspired by a large number of optical flow and stereo techniques [7][12][13],
 175 the superpixel flow is modelled with a pairwise Markov Random Field. The
 176 matching is performed via maximum-a-posteriori (MAP) inference on the la-
 177 beling l , which is equivalent to the minimization of an energy function of the
 178 form:

135
 136
 137
 138
 139
 140
 141
 142
 143
 144
 145
 146
 147
 148
 149
 150
 151
 152
 153
 154
 155
 156
 157
 158
 159
 160

161
 162
 163
 164
 165
 166
 167
 168
 169
 170
 171
 172
 173
 174
 175
 176
 177
 178
 179

180

$$E(l) = \sum_{p \in \Gamma} D_p(l_p; I_0, I_1) + \sum_{(p,q):q \in \mathcal{N}_r} S_{p,q}(l_p, l_q). \quad (1)$$

183

With l the set of labels of the superpixels in I_0 , that match with those in I_1 . \mathcal{N}_r is a neighbourhood of radius r of the superpixel p . The terms D , and S in (7) stand for data term and spatial smoothness term. The first one determines how accurate is the labeling in terms of consistency with the measured data (color, shape,etc.). In the classical optical flow equivalent of this equation, the data term corresponds to the pixel brightness conservation[7][5]. However, as superpixels are a set of similar (or somehow homogeneous) pixels, an adequate appearance based feature is a low dimensional color histogram (with N bins). So, here, D is the Hellinger distance between the histograms:

193

$$D_p(l_p; I_0, I_1) = \sqrt{1 - \frac{1}{\sqrt{\mathbf{h}(p)\mathbf{h}(p')N^2}} \sum_i \sqrt{\mathbf{h}_i(p)\mathbf{h}_i(p')}}. \quad (2)$$

196

Where $\mathbf{h}(p)$ and $\mathbf{h}(p')$ are the color histograms of the superpixel p and its correspondent superpixel in the second frame I_1 . For the experiments we used *RGB* color histogram with $N=3$ bins per color. Note that the low dimensional histogram gives certain robustness against noise, and slowly changing colors between frames.

201

On the other hand, the spatial term is a penalty function for spatial difference of the displacement vectors between neighboring superpixels, where a displacement vector has origin in the centroid of the superpixel of the first frame and end in the centroid of the superpixel of the second frame.

205

$$S_{p,q}(l_p, l_q) = \lambda(p) \sqrt{\frac{|u_{p_c} - u_{q_c}|}{\|p_c - q_c\|} + \frac{|v_{p_c} - v_{q_c}|}{\|p_c - q_c\|}}, \quad (3)$$

209

where, $\lambda(p) = (1 + \rho(\mathbf{h}(p), \mathbf{h}(q)))^2$.

210

The operator ρ is the Hellinger distance as used in the data term (8). The histogram distance is nonetheless computed between adjacent superpixels p and q , which belong to the first image. The superpixels centroids are noted as q_c and p_c , and u_* and v_* are the horizontal and vertical changes between centroids. This term has a smoothing effect in superpixels that belong to the same object. It has to be observed that when two close superpixels are different, thus, more probable to belong to different objects within the image, the term λ allows them to have matches that do not hold the smoothness prior with the same strength.

218

The Quadratic Pseudo-Boolean Optimization (QPBO) [3][4] is used to minimize the proposed energy function, by merging a set of candidate matches for every superpixel in the first frame. The candidate matches are generated by assuming a proximity prior. This means, every possible match should be inside a search radius in the second frame. Fig. 16 shows matching results for several datasets. Observe that the matches are correct even in difficult cases (bottom right).

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

224

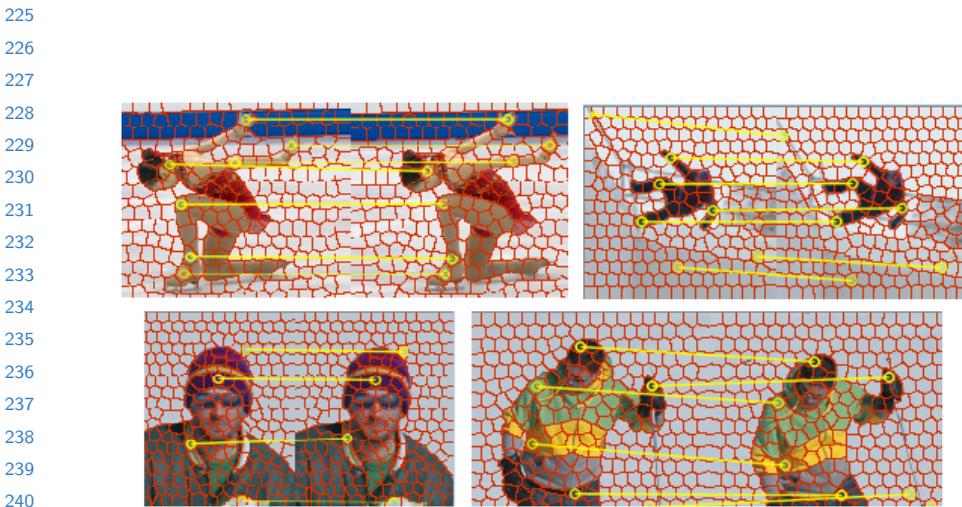


Fig. 4. The yellow lines show selected superpixel matching between pairs of images in several datasets.

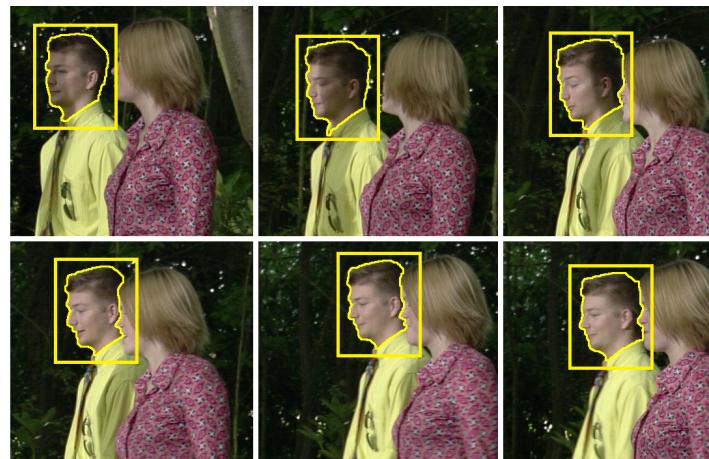
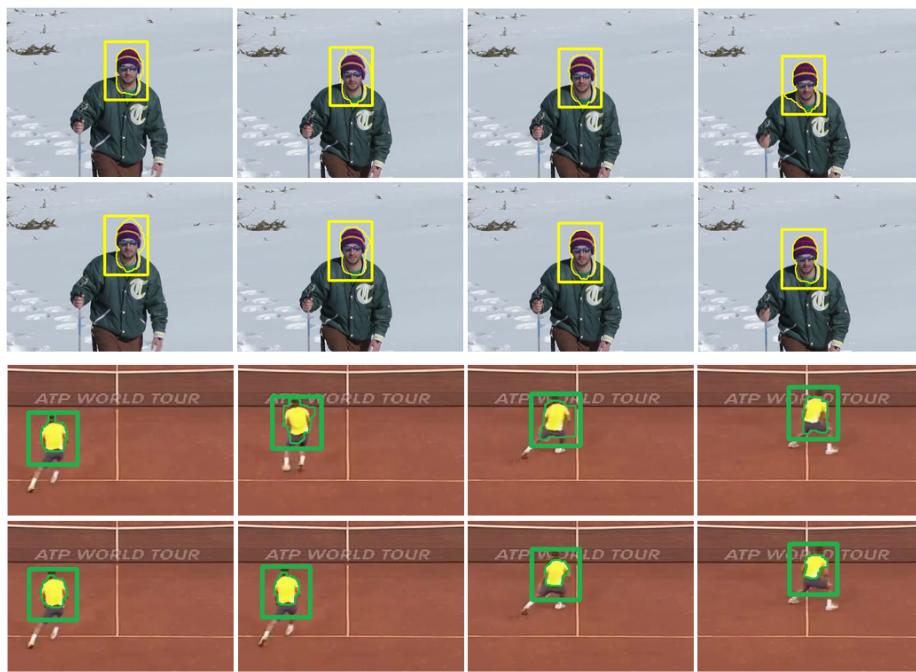


Fig. 5. Segmentation through the sequence Walking Couple (Yellow contour) initialized in the mans head. The yellow box correspond to the tracker output. The labeled background superpixel are not shown for clarity.

270 Fig. 17 shows segmentation results for an image sequence where the interest
 271 object is the head of a person. The head tracker and the superpixel flow provide
 272 information for better background-foreground separation. The method is tested
 273 in the Walking Couple sequence, by allowing only a small amount of iterations
 274 in the grab-cut segmentation. Observe how the contour in the man's head is
 275 correctly delineated when another person's head occludes part of it. In this case,
 276 the superpixels that belong to the womans face were correctly propagated and
 277 thus, labeled as background, despite the similar (skin) color between foreground
 278 and background regions.

279 In order to understand the effect of including superpixel propagation in a
 280 video sequence for object segmentation, some results are shown in the Fig. 18. For
 281 these experiments only one iteration is allowed in the graph-cut based methods.
 282 The top row frames (Fig. 18) were initialized only with the tracker, and the
 283 bottom row was initialized with the superpixel tracking technique. Observe that
 284 in general, the contour delineated is usually better in terms of precision and
 285 stability for the later one.



308 **Fig. 6.** Face segmentation in the Snow Shoes sequence and T-shirt extraction from
 309 Tennis sequence in several frames. For each group, the Top Row: One-iteration grab-cut
 310 initialized with tracker window; and the Bottom Row: One-iteration grab-cut initialized
 311 with the background regions tracking.

312

313 **Flow estimation.** The object flow consist on computing the motion field
 314 for an object of interest through an image sequence. The most usual approach

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

286

287

288

289

290

291

292

293

294

295

296

297

298

299

300

301

302

303

304

305

306

307

308

309

310

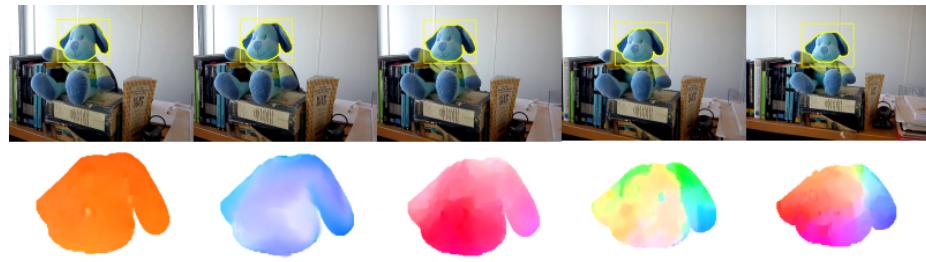
311

312

313

314

315 to solve a problem like this is to implement some of the available optical flow
 316 techniques through the complete sequence and perform the flow integration.
 317 However, this process results in high levels of motion drift [18][19] and usually
 318 the motion of the interest object is affected by a global regularization. In
 319 some extreme cases, the interest object motion may be totally blurred and other
 320 techniques have to be incorporated. Moreover, the diversity of natural video
 321 sequences makes difficult the choice of one technique over another, even when
 322 specialized databases are at hand [17], because currently no single method can
 323 achieve a strong performance in all of the available datasets. Most of these meth-
 324 ods consist in the minimization of an energy function with two terms (As was
 325 previously mentioned in this section). The data term is mostly shared between
 326 different approaches, but the prior or spatial term is different, and it basically
 327 states under what conditions the optical flow smoothness should be maintained
 328 or not. In a global approach, however, this is a difficult concept to define. Most
 329 of these smoothness terms rely in appearance differences or gradients. All these
 330 meaning that, unavoidably, some methods may be more reliable for some cases
 331 but weaker for others. It can be argued that this behaviour may be caused be-
 332 cause most of the techniques do not count with a way to identify firmly where
 333 exactly this smoothness prior can be applied.



345 **Fig. 7.** Object flow with the color code of [17] (bottom) for frames in the Puppy
 346 sequence (up).

347
 348 The main idea behind the object flow is that given the availability of several
 349 robust tracking techniques, and the proposed segmentation method for video, the
 350 optical flow computation can be refined by taking into account the segmentation
 351 mask within the tracked windows. The basic proposal to perform this refinement
 352 consist on considering the segmentation limits as reliable smoothness boundaries.
 353 This is, of course, under the assumption that the motion is indeed smooth within
 354 the object region. This assumption is not far from reality in most scenes
 355 with an interest object. Naturally, as the object tracker is included, is expected
 356 that the object flow should be more robust to rapid motions than the optical
 357 flow. Thus, the full motion is split in two, the long range motion, given by the
 358 tracker window, and the precision part, given by the targeted optical flow. The
 359 Fig. 19 shows the object flow for a frame in the Puppy sequence. Observe the

315
 316
 317
 318
 319
 320
 321
 322
 323
 324
 325
 326
 327
 328
 329
 330
 331
 332
 333
 334
 335
 336
 337
 338
 339
 340
 341
 342
 343
 344
 345
 346
 347
 348
 349
 350
 351
 352
 353
 354
 355
 356
 357
 358
 359

360 motion vectors are computed only inside the object of interest, preserving a
 361 strong smoothing prior, but also allowing internal variations in the flow.

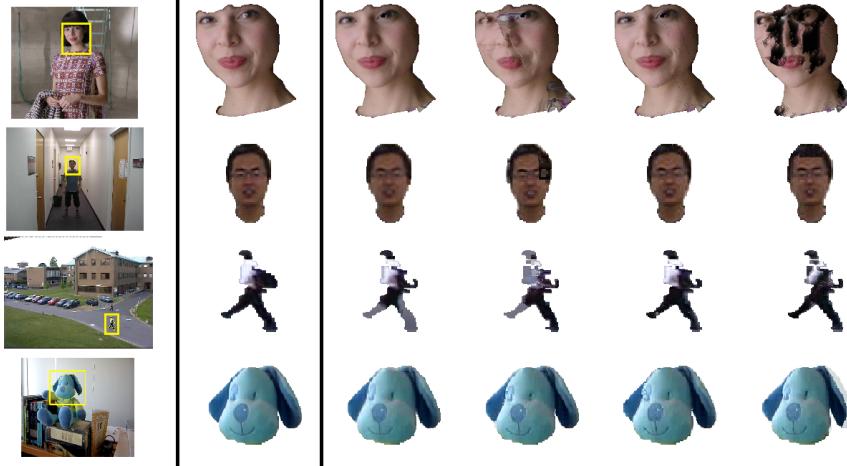
362 As a first approximation to the object flow, the Simple Flow technique [21]
 363 is taken as core base. This is because of its scalability to higher resolutions and
 364 because its specialization to the concept of object flow is only natural. This is
 365 because in the Simple Flow pipeline the smoothness localization can be easily
 366 specified through computation masks. More specifically, the initial computation
 367 mask is derived from the segmentation performed as prior step. The resulting
 368 flow is then filtered only inside the mask limits to enhance precision and fasten-
 369 ing the implementation. However, direct modifications in other optical flow
 370 methods can be further studied. For instance, in graph-cut based minimization
 371 approaches, the regularity constraints can be precisely targeted by disconnecting
 372 foreground pixels from background ones.

373

374 3 Experimental results

375

376



377
 378
 379
 380
 381
 382
 383
 384
 385
 386
 387
 388
 389
 390
 391
Fig. 8. Extrapolation results from integrated flow in 4 sequences. In descending order:
 392 Amelia Retro, Boy, Walking, Puppy. From Left to Right: Annotated object, Backward
 393 object flow, Backward optical flow, Forward object flow, Backward optical flow.
 394

395

396 To evaluate the performance of the object flow in comparison with opti-
 397 cal flow techniques, we performed a number of experiments on several video
 398 sequences. We annotated an initial bounding box for the videos, and a segmen-
 399 tation contour of the interest object for every frame. The experiment measures
 400 the ability of the method to extrapolate an image from the initial frame and
 401 the integrated flow. For every pair of frames the video sequence, the PSNR be-
 402 tween the annotated current state of the object and the extrapolated images is
 403 computed. The Fig. 20 is a sample of the performed experiment, each column
 404 is an image generated from the given flow. Two types integration are evaluated,

396

397

398

399

400

401

402

403

404

360

361

362

363

364

365

366

367

368

369

370

371

372

373

374

375

376

377

378

379

380

381

382

383

384

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

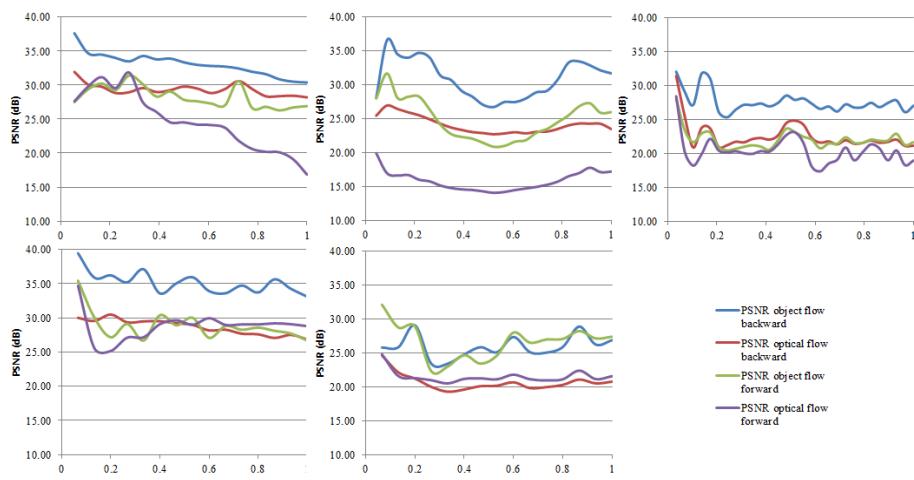
402

403

404

405 *From-the-reference*, or forward integration, and *To-the-reference*, or backward
 406 integration, as discussed in [20]. So, for each row in the Fig. 20, two columns
 407 correspond to the object flow, and two columns correspond to optical flow, with
 408 both types of integration.

409 The Fig. 21 shows PSNR graphics for 4 different sequences. For every pair
 410 of frames an image is extrapolated, and the PSNR with the ground-truth object
 411 is computed. The results are shown with both, Euler integration (Labelled as
 412 *forward* in the figs.) of the used flow, and using the integration method described
 413 in [20], labeled as *backward* in the figures. The results show that the object flow
 414 methods are generally more precise than its optical flow counterparts. Moreover,
 415 the object flow method with backward integration usually performs much better
 416 than any other combination of techniques. For this experiment, the object flow
 417 is compared with the simple-flow optical-flow method.
 418



435 **Fig. 9.** PSNR graphs for extrapolated images using Object flow and the Simple Optical
 436 Flow for 4 sequences. In descendent order: Puppy Seq.; Amelie Retro Seq.; Boy Seq.;
 437 Walking Seq.

438
 439
 440 The Fig. 22 presents a visual comparison between the object flow and several
 441 optical flow techniques in the Amelia sequence for object extrapolation, and the
 442 involved frames (the first and last used frames in the sequence). The Fig. 23
 443 shows the PSNR results for every extrapolated frame in the full sequence, the
 444 object flow performs better than all the studied optical flow techniques.
 445

446 Observe that the object details are lost in comparison with the ground-truth
 447 object image (Fig. 22). For example, the closed eyes detail is missing in the most
 448 of the optical flow methods. Furthermore, several of the methods lost any sig-
 449 nificance, and the output barely holds any resemblance with the original image.



Fig. 10. Top: Comparison between extrapolated objects using several methods: Groundtruth object, Object flow, TVL1, Block Matching, Brox, Farneback and Simple Flow. Bottom: First and current frame. The extrapolation is performed using backward accumulation of the flow.

3.1 Object flow for video edition

Patch replacement is a common task in video post-production, and it usually requires an exhausting and time consuming frame-by-frame edition. The time to make such modification can be widely reduced by implementing the object flow, and taking the interest zone as the target object. The edition can be done only in a couple of keyframes, and then, an automatic expansion to the rest of the frames can be performed. Fig. 24 shows the proposed method. The top row contains the automatically edited frames (logo insertion), and the bottom row shows the original frames. Pay attention to the difficult non-rigid transformations.

4 Conclusions

A framework to combine tracking and optical flow methods to improve object based dense motion description is presented. The pipeline is composed of three main steps, object tracking, segmentation and flow estimation. For the segmentation step a new promising video object segmentation algorithm was proposed, and, to the best of our knowledge, the introduced superpixel flow is the first energy based algorithm for superpixel matching. For the last step, we presented a flow estimation method based on a modification of the simple-flow method to use the obtained segmentation mask. The experiments showed that this object based flow estimation improves the dense motion estimation in comparison to optical flow techniques. Future work can be further explored in the use of the object flow as feedback hint for tracking-by-detection methods. Also, several kind

CONFIDENTIAL REVIEW COPY. DO NOT DISTRIBUTE.

12 ACCV-14 submission ID ***

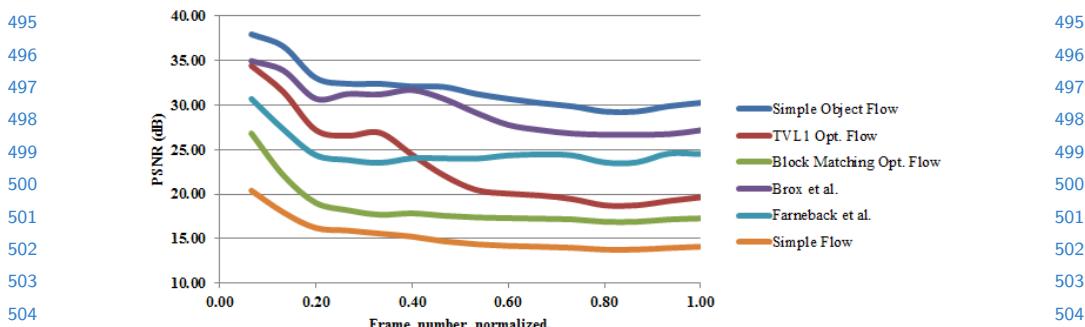


Fig. 11. PSNR graphs for extrapolated images using Object flow and the different Optical Flow techniques for the Amelia sequence.

of applications of the object flow can be more deeply approached. For instance, in the structure-from-motion pipeline, video based rendering, automatic video edition, and video inpainting among others.

References

1. J. Malik and X. Ren, Learning a classification model for segmentation, *Computer Vision, International Conference*, 2003.
2. S. Boltz; F. Nielsen and S. Soatto, Earth mover distance on superpixels, *International Conference on Image Processing*, 2010.
3. E. Boros; P. Hammer and G. Tavares, Preprocessing of unconstrained quadratic binary optimization, *RUTCOR*, 2010.
4. E. Boros and P. Hammer, Pseudo-boolean optimization, *Discrete applied Mathematics*, 2002.
5. B. Horn and B. Schunck, Determining Optical Flow, *Artificial Intelligence*, 1981.
6. H. Ishikawa and P. Bouthemy, Multimodal estimation of discontinuous optical flow using Markov random fields, *TPAMI*, 1993.
7. V. Lempitsky, S. Roth and C. Rother, Fusion Flow: Discrete-Continuos optimization for optical flow estimation, *Computer Vision and Pattern Recognition*, 2008.



Fig. 12. An application of the object flow for augmented reality/video edition.

- 540 8. M. Reso and J. Jachalsky, Temporally Consistent Superpixels, *International Conference Computer Vision.*, 2011. 540
 541 9. R. Achanta; A. Shaji; K. Smith; Aurelien Lucchi; P. Fua and S. Susstrunk, SLIC 541
 542 Superpixels compared to state of the art superpixel methods, *Discrete applied Mathematics.*, 2002. 542
 543 10. F. Perbet and A. Maki, Homogeneous superpixels from random walks, *MVA.*, 2011. 543
 544 11. C. Xu and J.J. Corso. Evaluation of super-voxel methods for early video processing, 544
 545 *Computer Vision and Pattern Recognition.* 2012. 545
 546 12. A. Shekhovtsov, I. Kovtun and V. Hlavac. Efficient MRF deformation model for 546
 547 non-rigid image matching, *Computer Vision and Pattern Recognition.* 2007. 547
 548 13. J. Sun, N.N Shen and H.Y. Shum. Stereo matching using propagation belief, 548
 549 *TPAMI.* 2003. 549
 550 14. C. Rother, V. Kolmogorov and A. Blake. Grabcut: Interactive foreground extraction 550
 551 using iterated graph cuts, *SIGGRAPH.* 2004. 551
 552 15. L. Yang, Y. Guo, X. Wu and X. Wang. A new video segmentation approach: 552
 553 Grabcut in local window, *Soft Computing and Pattern Recognition.* 2011. 553
 554 16. Y. Wu, J. Lim and M.-H. Yang. Online object tracking: A benchmark, *Computer 554
 555 Vision and Pattern Recognition.* 2013. 555
 556 17. S. Baker, D. Scharstein, J.P. Lewis, S. Roth, M.J. Black and R. Szeliski. A Database 556
 557 and Evaluation Methodology for Optical Flow, *International Journal Computer 557
 558 Vision.* 2013. 558
 559 18. Y. Boykov, M-P. Jolly. Interactive Graph Cuts for Optimal Boundary & Region 559
 560 Segmentation of Objects in N-D images, *International Conference on Computer 560
 561 Vision.* 2013. 561
 562 19. W. Li, D. Cosker and M. Brown. An anchor patch based optimization framework for 562
 563 reducing optical flow drift in long image sequences, *Asian Conference on Computer 563
 564 Vision.* 2012. 564
 565 20. T. Crivelli, P.-H. Conze, P. Robert, M. Fradet and P. Perez. Multi-step flow fusion: 565
 566 towards accurate and dense correspondence in long video shots, *British Conference 566
 567 Machine Vision.* 2012. 567
 568 21. M. Tao, J. Bai, P. Kohli, and S. Paris. SimpleFlow: A Non-iterative, Sublinear 568
 569 Optical Flow Algorithm, *Computer Graphics Forum, Eurographics.* 2012. 569
 570 22. Brox, A. Bruhn, N. Papenberg, J. Weickert. High accuracy optical flow estimation 570
 571 based on a theory for warping. *European Conference in Computer Vision.* 2004. 571
 572 23. S. Hare, A. Saffari, and P.H.S. Torr, Struck: Structured Output Tracking with 572
 573 Kernels. *International Conference on Computer Vision.* 2011. 573
 574 24. B. Babenko, M.H. Yang, and S. Belongie. Visual Tracking with Online Multiple 574
 575 Instance Learning. *Computer Vision and Pattern Recognition.* 2009. 575
 576 25. J. Shi and C. Tomasi. Good features to track. *Conference on Computer Vision and 576
 577 Pattern Recognition ,* 1994. 577
 578
 579
 580
 581
 582
 583
 584