

745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806

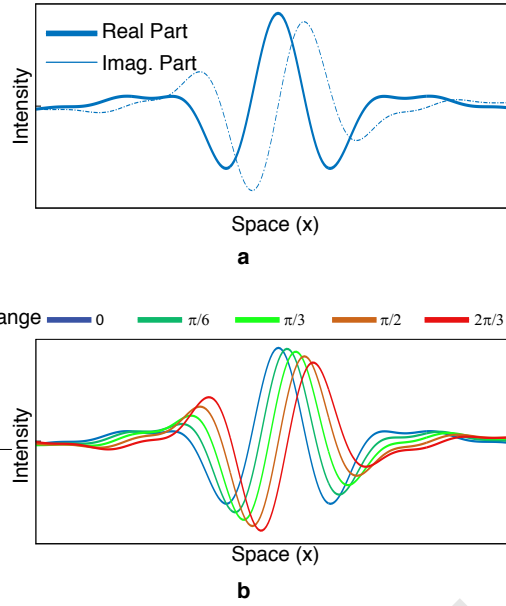


Fig. S1. Increasing the phase of complex steerable pyramid coefficients results in approximate local motion of the basis functions. **a.** A 1D slice of a complex steerable pyramid basis function. **b.** The basis function is multiplied by several complex coefficients of constant amplitude and increasing phase to produce the real part of a new basis function that is approximately translating. (Reproduced from (31).)

807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868

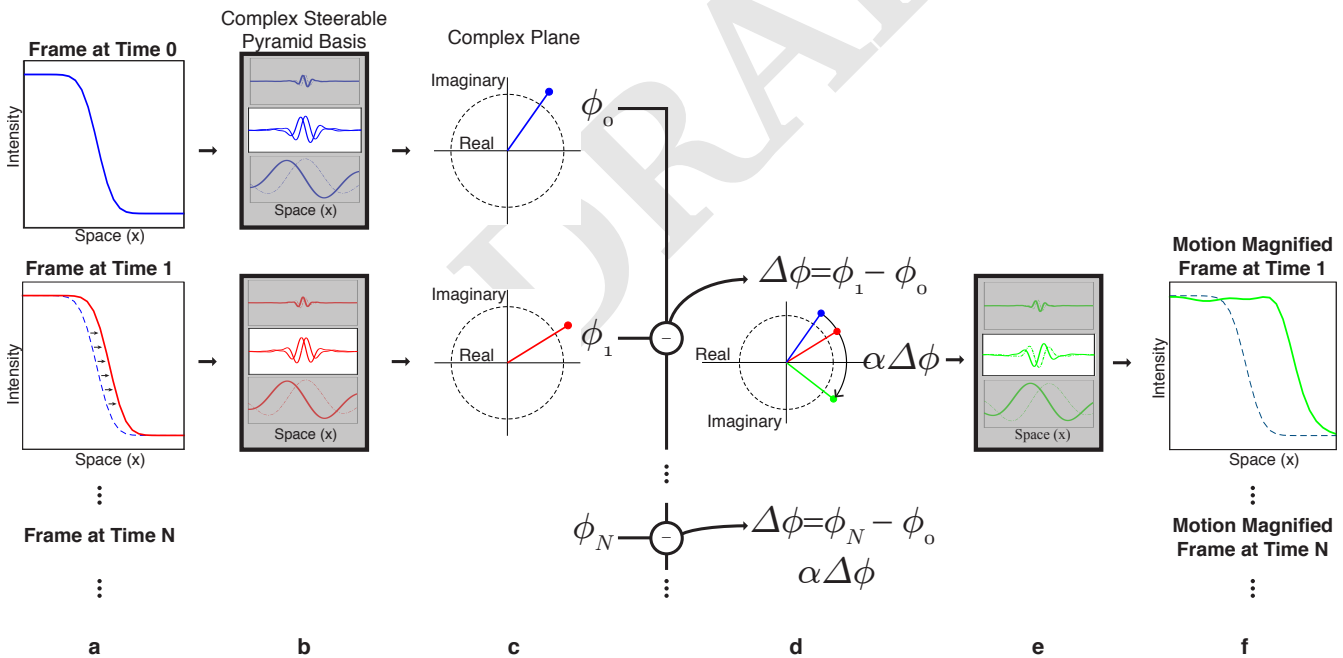


Fig. S2. A 1D example illustrating how the local phase of complex steerable pyramid coefficients is used to amplify the motion of a subtly translating step edge. **a.** Frames (two shown) from the video. **b.** Sample basis functions of the complex steerable pyramid. **c.** Coefficients (one shown per frame) of the frames in the complex steerable pyramid representation. The phases of the resulting complex coefficients are computed. **d.** The phase differences between corresponding coefficients are amplified. Only a coefficient corresponding to a single location and scale is shown; this processing is done to all coefficients. **e.** The new coefficients are used to shift the basis functions. **f.** A reconstructed video is produced by inverse transforming the complex steerable pyramid representation. The motion of the step edge is magnified. (Reproduced from (31).)

869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930

931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992

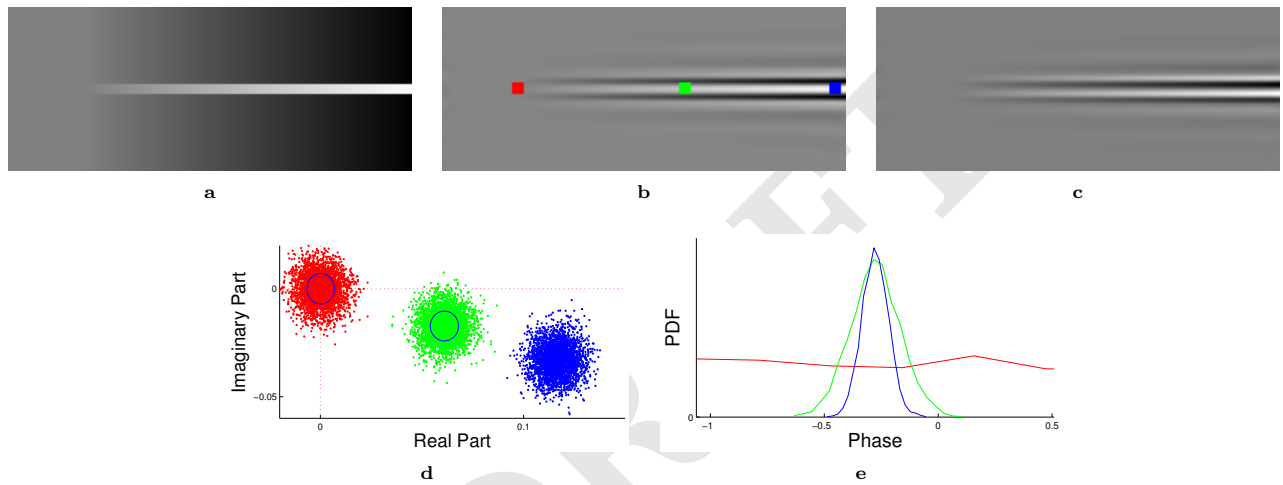


Fig. S3. Noise model of local phase. **a**, A frame from a synthetic video with noise. **b**, The real part of a single level of the complex steerable pyramid representation of **a**. **c**, The imaginary part of the same level of the complex steerable pyramid representation of **a**. **d**, A point cloud over noisy values of the real and imaginary values of the complex steerable pyramid representation at the red, green and blue points in **b**. **(e)**. The corresponding histogram of phases.

993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054

1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116

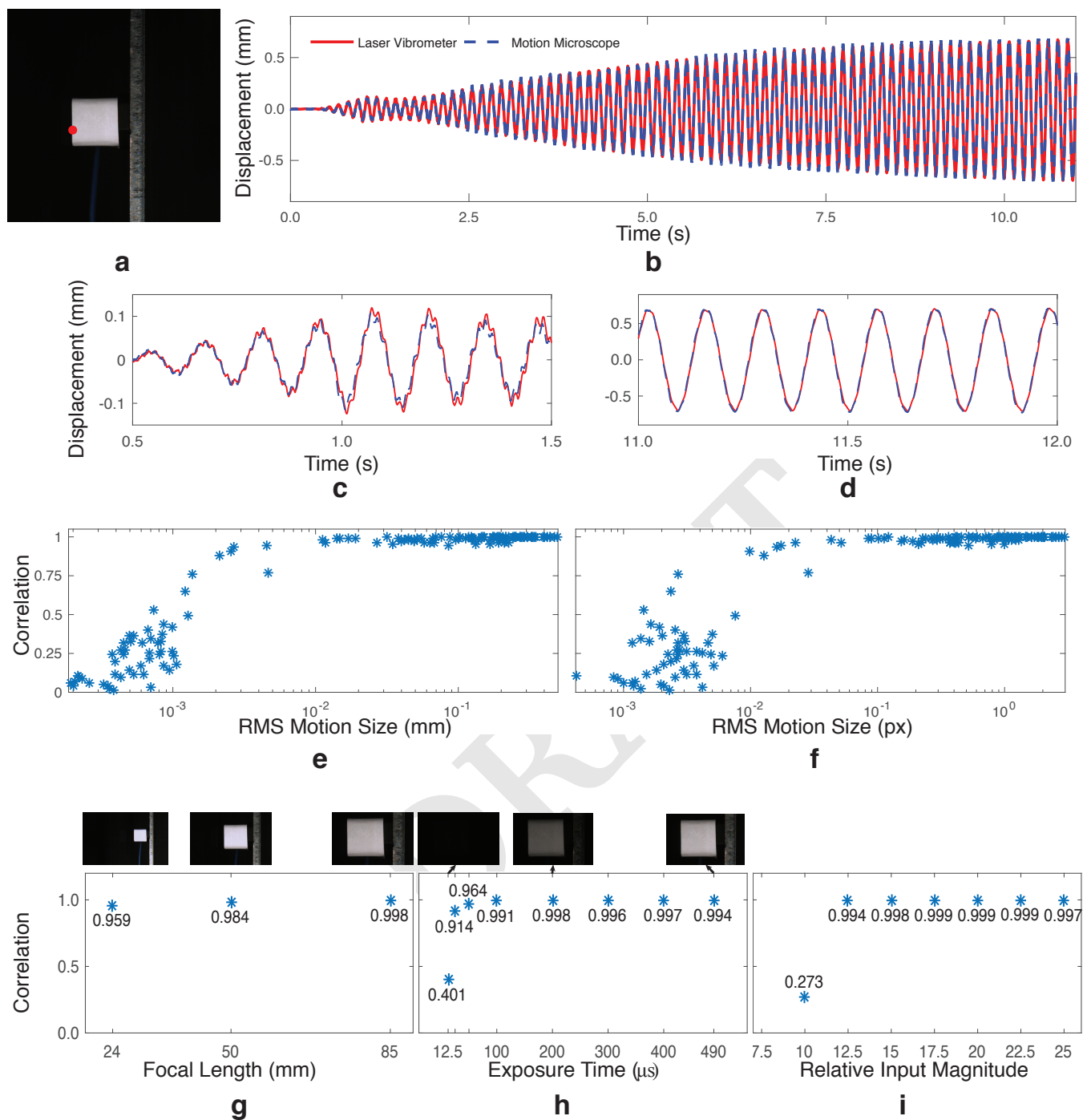


Fig. S4. A comparison of our quantitative motion estimation vs. a laser vibrometer. Several videos of a cantilevered beam excited by a shaker were taken with varying focal length, exposure times and excitation magnitude. **a**, A frame from one video. **b**, Motions from the motion microscope at the red point are compared to the integrated velocities from a laser vibrometer. **c**, **b** from 0.5-1.5s. **d**, **b** from 11-12s. **e**, The correlation between the two signals across the videos vs. the root-mean-square (RMS) motion size in millimeters measured by the laser vibrometer. **f**, The correlation between the two signals across the videos vs. the root-mean-square (RMS) motion size in pixels measured by the laser vibrometer. **g**, The correlation between the signals vs. focal length (exposure time: 490 μ s, excitation magnitude: 15). **h**, Correlation vs. exposure time (focal length: 85mm, excitation magnitude: 15). Cropped frames from the corresponding videos are shown above. **i**, Correlation vs. relative excitation magnitude (focal length: 85 mm, exposure time: 490 μ s). Only motions at the red point in **a** were used in our analysis.

1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178

1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240

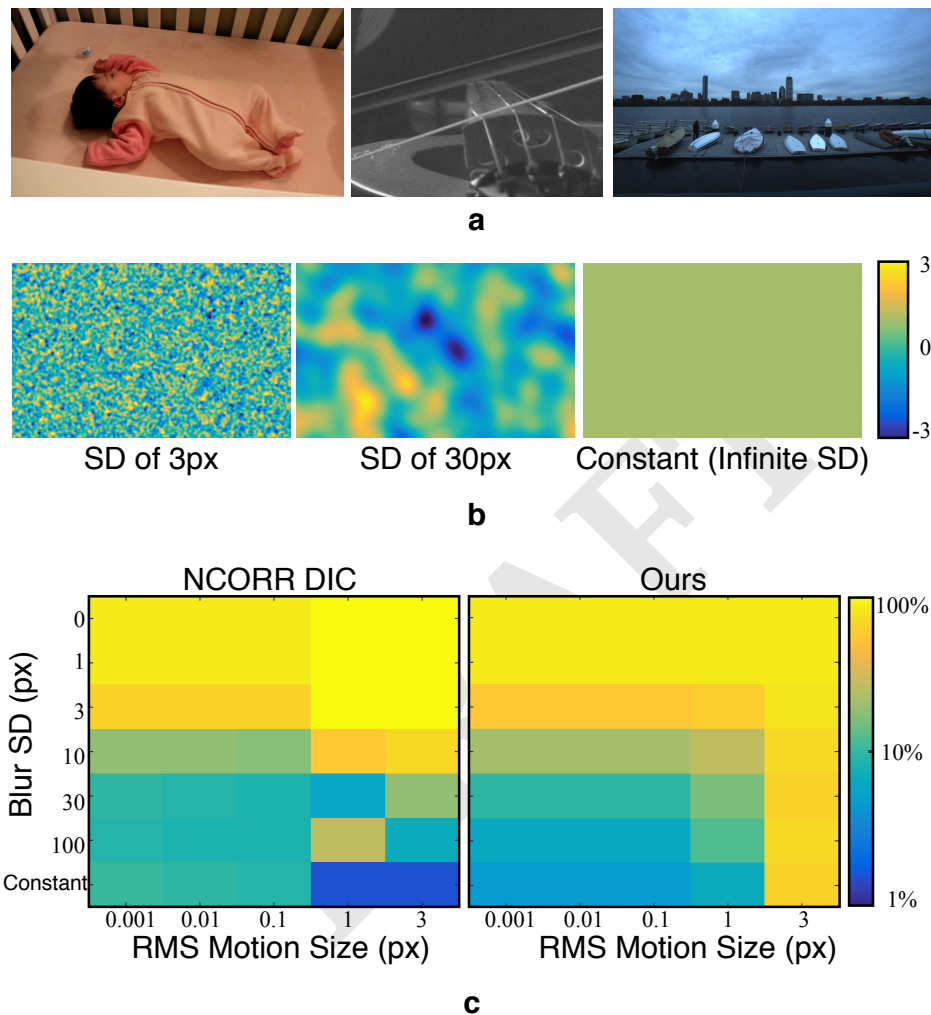


Fig. S5. An evaluation of our motion estimation method and NCORR (32) on a synthetic dataset of images. **a**, Sample frames from real videos used to create dataset. **b**, Sample of synthetic motion fields of various motion size and spatial scale used to create dataset. **c**, The motion microscope and NCORR are used to estimate the motion field and the average relative error is displayed for both methods as a function of motion size and spatial scale. Both methods are only accurate for spatially smooth motion fields. Our method is twice as accurate for spatially smooth, sub-pixel motion fields. NCORR is more accurate for larger motions.

1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302

1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364

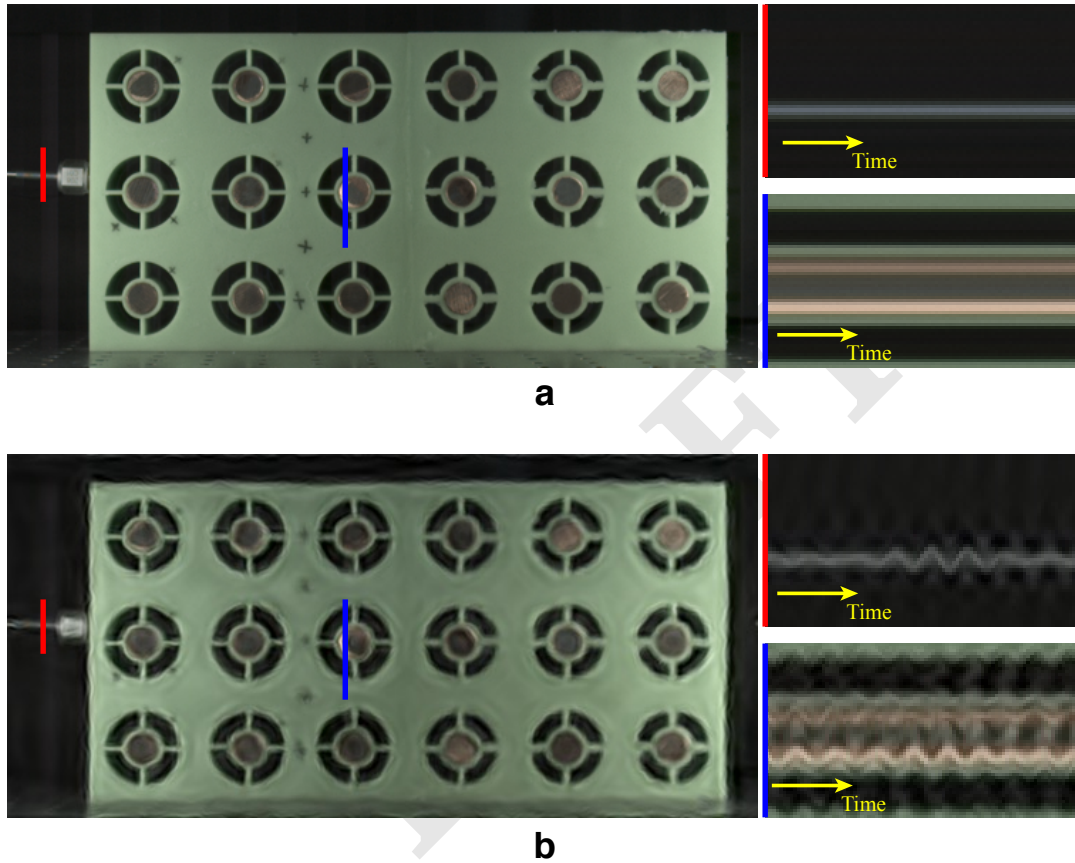


Fig. S6. Magnification of a spatial smooth and temporally filtered noise can look like real signal. **a**, Frames and time slices from a synthetic 300-frame video created by replicating a single frame 300 times and adding a different realistic noise pattern to each frame. **b**, Corresponding frame and time slices from the synthetic video motion magnified 600x in a temporal band of 40-60Hz. (b). Timeslices from the same parts of each video are shown on the right for comparison.

1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426

1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488

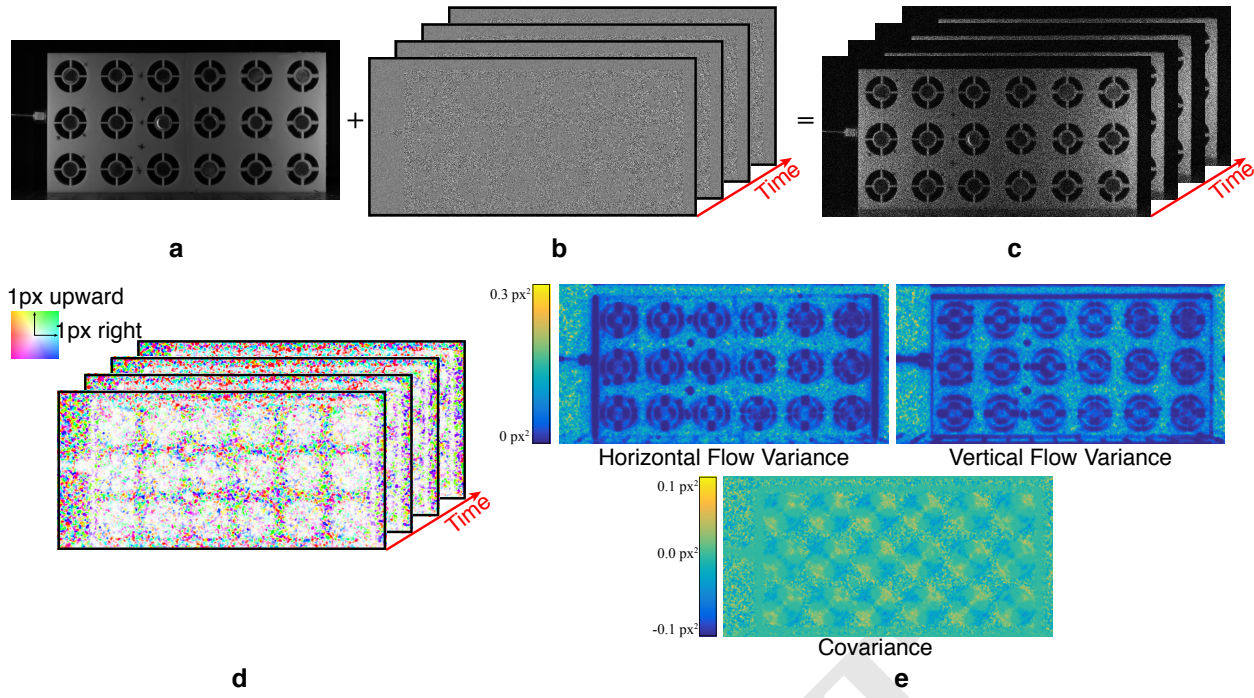


Fig. S7. Using a probabilistic simulation to compute the noise covariance of the motion estimate. **a**, A single frame from an input video, in this case of an elastic metamaterial. **b**, Simulated, but realistic noise (contrast enhanced 80x). **c**, Synthetic video with no motions consisting of the input frame replicated plus simulated noise (noise contrast-enhanced 80x). **d**, Estimated motions of this video. **e**, Sample variances and sample covariance of the vertical and horizontal components of the motion are computed to give an estimate of how much noise is in the motion estimate.

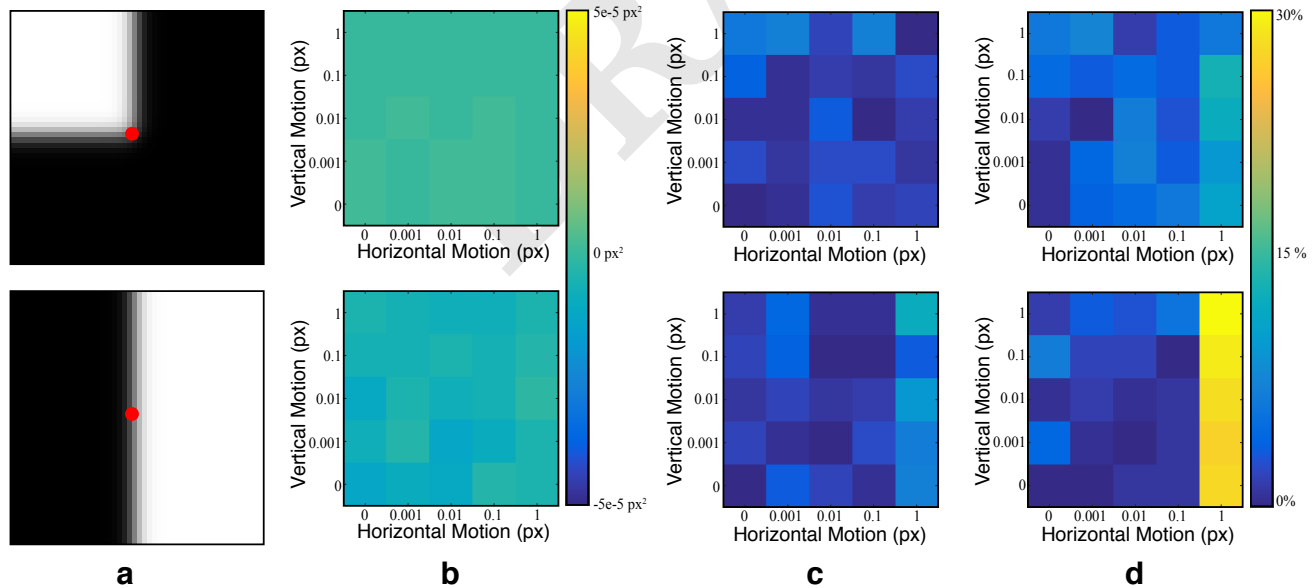


Fig. S8. Synthetic experiments showing that our noise covariance estimation, which assumes that the motions are zero, is also accurate for small non-zero motions. **a**, The motion between synthetic frames with noise and slightly translated versions (not shown) are computed over 4000 runs at the marked point in red for several different translation amounts. Each time different, but independent noise is added to the frames. **b**, The sample covariance vs. motion size. **c**, Relative error of horizontal variance vs. motion size. **d**, Relative error of vertical variance vs. motion size. **c** and **d** are on the same color scale.

1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550

1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612

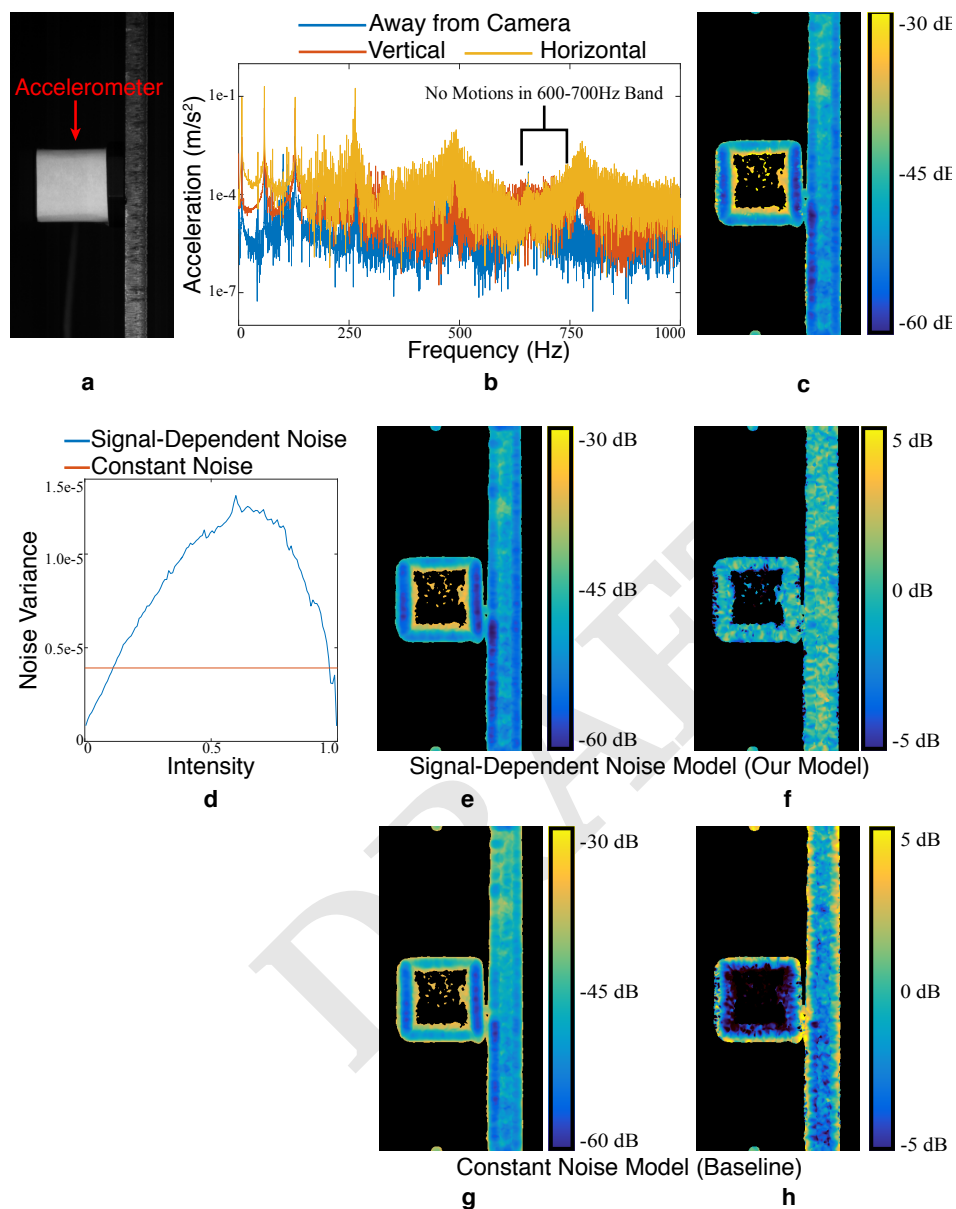


Fig. S9. Validation of our noise estimation on real data. **a**, A frame from a real video of an accelerometer attached to a beam. **b**, The accelerometer shows there are no motions in the frequency band 600 to 700 Hz. **c**, The variance of our motion estimate in the 600-700Hz band serves as a ground truth measure of noise as there are no motions. **d**, The estimated noise level vs. intensity for a signal-dependent noise model and a constant noise model. **e**, The noise estimate produced by our Monte Carlo simulation with a signal-dependent model. All variances are of the motions projected onto the direction of least variance. Textureless regions, where the motion estimation is not meaningful, have been masked out in black. **f**, Difference in decibels between ground truth and **e**. **g**, Noise estimate produced by the Monte Carlo simulation with a constant noise model. **h**, Difference in decibels between ground truth and **g**.

1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674

1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736

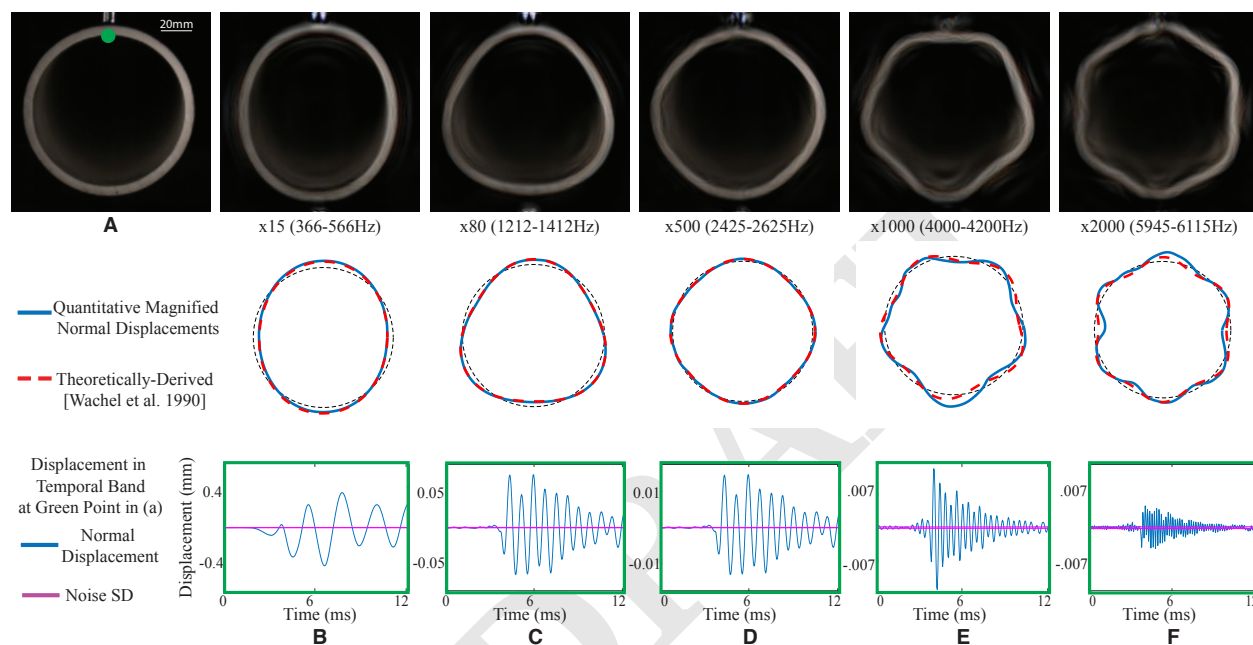


Fig. S10. The motion microscope is applied to a pipe being struck by a hammer. (A) A frame from the input video, recorded at 24,096 frames per second. (B-F) In the first row, a frame is shown from five motion magnified videos showing motions amplified in the specified frequency bands (Supplementary Video 3). In the second row, modal shapes recovered from a quantitative analysis of the motions are shown in blue. The theoretically-derived modal shapes, shown in dashed orange, are overlaid for comparison over a perfect circle in dotted black. In the bottom row, displacement vs. time and the estimated noise standard deviation is shown at the green point marked in A.

1737	Supporting Information	1799
1738		1800
1739	1. Modal Shapes of a Pipe	1801
1740		1802
1741		1803
1742	We made a measurement of a pipe being struck by a hammer, viewed end on by a camera, to capture its radial-circumferential vibration	1804
1743		1805
1744	modes. A standard 4" schedule 40 PVC pipe was recorded with a high-speed camera at 24,000 frames per second (fps), at a resolution of	1806
1745		1807
1746	192 × 192 (SI Appendix, Fig. S10a). SI Appendix, Fig. S10b-f shows frames from the motion magnified videos for different resonant	1808
1747		1809
1748	frequencies showing the mode shapes, a comparison of the quantitatively measured mode shapes with the theoretically derived mode	1810
1749		1811
1750	shapes, and the displacement vs. time of the specific frequency band and the estimated noise standard deviation. The tiny modal	1812
1751		1813
1752	motions are seen clearly. Obtaining vibration data with traditional sensors with the same spatial density would be extremely difficult, and	1814
1753		1815
1754	accelerometers placed on the pipe would alter its resonant frequencies.	1816
1755		1817
1756		1818
1757	This sequence also demonstrates the accuracy of our noise analysis. The noise standard deviations show that the detected motions	1819
1758		1820
1759	prior to impact, when the pipe is stationary, are likely spurious.	1821
1760		1822
1761		1823
1762	2. Synthetic Validation	1824
1763		1825
1764		1826
1765	We validate the accuracy of our motion estimation on a synthetic dataset and compare its accuracy to NCORR, a digital image correlation	1827
1766		1828
1767	technique (32) used by mechanical engineers (33). In this experiment, we did not employ temporal filtering.	1829
1768		1830
1769	We created a synthetic dataset of frame pairs with known ground truth motions between them. We took natural images from the	1831
1770		1832
1771	frames of real videos (SI Appendix, Fig. S5a) and warped them according to known motion fields using cubic b-spline interpolation (34).	1833
1772		1834
1773	Sample motions fields, shown in SI Appendix, Fig. S5b, were produced by Gaussian blurring IID Gaussian random variables. We used	1835
1774		1836
1775	Gaussian blurs with standard deviations (SD), ranging from zero (no filtering) to infinite (a constant motion field). We also varied the	1837
1776		1838
1777	root-mean-square (RMS) amplitude of the motion fields from 0.001px to 3px. For each set of motion field parameters, we sampled five	1839
1778		1840
1779		1841
1780	different motion fields to produce a total of 155 motion fields with different amplitudes and spatial coherence. To test the accuracy of the	1842
1781		1843
1782	algorithms rather than their sensitivity to noise, no noise was added to the image pairs.	1844
1783		1845
1784		1846
1785	We ran our motion estimation technique and NCORR on each image pair. We then computed the mean absolute difference between	1847
1786		1848
1787	the estimated and ground truth motion fields. Then, for each set of motion field parameters, we averaged the mean absolute differences	1849
1788		1850
1789	across image pairs and divided the result by the RMS motion amplitude to make the errors comparable over motion sizes. The result is	1851
1790		1852
1791	the average relative error as a percentage of RMS motion amplitude (SI Appendix, Fig. S5c).	1853
1792		1854
1793	Both NCORR and our method perform best when the motions are spatially coherent (filter standard deviations greater than 10 px)	1855
1794		1856
1795	with relative errors under 10%. This reflects the fact that both methods assume the motion field is spatially smooth. Across motion sizes,	1857
1796		1858
1797		1859
1798	our method performs best for sub-pixel motions (5% relative error). This is probably because we assume that the motions are small when	1860

1861 we linearize the phase constancy equation (Eq. 9). NCORR has twice the relative error (10%) for the same motion fields. 1923
1862 1924
1863 The relative errors reported in SI Appendix, Fig. S5c are computed over all pixels including those that are in smooth, textureless 1925
1864 regions where it is difficult to estimate the motions. If we restrict the error metric to only take into account pixels at edges and corners, 1926
1865 the average relative errors for small ($< 1\text{px}$ RMS), spatially coherent (filter SD $> 10\text{px}$) motions drops by a factor of 2.5 for both methods. 1927
1866 1928
1867 1929
1868 the average relative errors for small ($< 1\text{px}$ RMS), spatially coherent (filter SD $> 10\text{px}$) motions drops by a factor of 2.5 for both methods. 1930
1869 1931
1870 We generated synthetic images that are slight translations of each other and added Gaussian noise to the frames (SI Appendix, Fig. S8a). 1932
1871 1933
1872 For each translation amount, we compute the motion between the two frames over 4000 runs. We compute the sample covariance matrix 1934
1873 over the runs as a measure of the ground truth noise level. We also used our noise analysis to estimate the covariance matrix at the points 1935
1874 denoted in red. 1936
1875 1937
1876 1938
1877 1939
1878 The off-diagonal term of the covariance matrix should be zero for the synthetic frames in SI Appendix, Fig. S8a. For both examples, it 1940
1879 is within 10^{-5}px^2 of zero for all translation amounts (SI Appendix, Fig. S8b). 1941
1880 1942
1881 1943
1882 1944
1883 The relative errors of the horizontal and vertical variances vs. translation (SI Appendix, Fig. S8c-d) are less than 5% for sub-pixels 1945
1884 motions. This is likely due to the random nature of the simulation. For motions greater than one pixel, the covariance matrix has relative 1946
1885 error of less than 25%. 1947
1886 1948
1887 1949
1888 1950
1889 **3. Relation between Local Phase Differences and Motions** 1951
1890 1952
1891 Fleet and Jepson have shown that contours of constant phase in image subbands such as those in the complex steerable pyramid 1953
1892 approximately track the motion of objects in a video (7). We make a similar *phase constancy* assumption, in which the following equation 1954
1893 relates the phase of the frame at time 0 to the phase of future frames: 1955
1894 1956
1895 1957
1896 1958
1897 1959
1898
$$\phi_{r,\theta}(x, y, 0) = \phi_{r,\theta}(x - u(x, y, t), y - v(x, y, t), t), \quad [7] \quad 1960$$
1899 1961
1900 where $\mathbf{V}(x, y, t) := (u(x, y, t), v(x, y, t))$ is the motion we seek to compute. We Taylor-expand the right-hand side around (x, y) to get 1962
1901 1963
1902
$$\Delta\phi_{r,\theta} = \left(\frac{\partial\phi_{r,\theta}}{\partial x}, \frac{\partial\phi_{r,\theta}}{\partial y} \right) \cdot (u, v) + O(u^2, v^2), \quad [8] \quad 1964$$
1903 1965
1904 1966
1905 1967
1906 where $\Delta\phi_{r,\theta}(x, y, t) := \phi_{r,\theta}(x, y, t) - \phi_{r,\theta}(x, y, 0)$, arguments have been suppressed and $O(u^2, v^2)$ represents higher-order terms in the 1968
1907 Taylor expansion. Because we assume the motions are small, higher order terms are negligible and the local phase variations are 1969
1908 approximately equal to only the linear term: 1970
1909 1971
1910
$$\Delta\phi_{r,\theta} = \left(\frac{\partial\phi_{r,\theta}}{\partial x}, \frac{\partial\phi_{r,\theta}}{\partial y} \right) \cdot (u, v). \quad [9] \quad 1972$$
1911 1973
1912 1974
1913 1975
1914 1976
1915 Fleet has shown that the spatial gradients of the local phase, $\left(\frac{\partial\phi_{r,\theta}}{\partial x}, \frac{\partial\phi_{r,\theta}}{\partial y} \right)$, are roughly constant within a subband and that they are 1977
1916 approximately equal to the peak tuning frequency of the corresponding subband's filter (35). This frequency is a 2D vector oriented 1978
1917 orthogonal to the direction the subband selects for, which means that the local phase changes only provide information about the motions 1979
1918 perpendicular to this direction. 1980
1919 1981
1920 1982
1921 1983
1922 1984

1985	4. Low-Amplitude Coefficients have Noisy Phase	2047
1986		2048
1987		2049
1988	Each frame of the input video $I(x, y, t)$ is transformed to the complex steerable pyramid representation by being spatially bandpassed by	2050
1989		2051
1990	a bank of quadrature pairs of filters $g_{r,\theta}$ and $h_{r,\theta}$, where r corresponds to different spatial scales of the pyramid and θ corresponds to	2052
1991		2053
1992	different orientations. We use the filters of Portilla and Simoncelli, which are specified and applied in the frequency domain (26). For one	2054
1993		2055
1994	such filter pair, the result is a set of complex coefficients $S_{r,\theta} + iT_{r,\theta}$ whose real and imaginary part are given by	2056
1995		2057
1996		2058
1997	$S_{r,\theta} = g_{r,\theta} * I \text{ and } T_{r,\theta} = h_{r,\theta} * I$	[10] 2059
1998		2060
1999		2061
2000	where the convolution is applied spatially at each time instant t . This filter pair is converted to amplitude $A_{r,\theta}$ and phase $\phi_{r,\theta}$ by the	2062
2001		2063
2002	operations	2064
2003		2065
2004	$A_{r,\theta} = \sqrt{S_{r,\theta}^2 + T_{r,\theta}^2} \text{ and } \phi_{r,\theta} = \tan^{-1}(T_{r,\theta}/S_{r,\theta})$	[11] 2066
2005		2067
2006		2068
2007	$g_{r,\theta}$ and $h_{r,\theta}$ are in quadrature relationship, which means that they select for the same frequencies, but are 90 degrees out of phase	2069
2008		2070
2009	like sin and cos. A consequence is that they are uncorrelated and have equal root mean square (RMS) value. Complex coefficients at	2071
2010		2072
2011	antipodal orientations are conjugate symmetric and contain redundant information. Therefore, we only use a half circle of orientations.	2073
2012		2074
2013		2075
2014	This transform has various properties that we don't use in this work such as perfect invertibility and steerability. Invertibility is used	2076
2015		2077
2016	in motion magnification.	2078
2017		2079
2018	Suppose the observed video $I(x, y, t)$ is contaminated with independent and identically distributed (iid) noise $I_n(x, y, t)$ of variance σ^2 :	2080
2019		2081
2020		2082
2021	$I(x, y, t) = I_0(x, y, t) + I_n(x, y, t)$	[12] 2083
2022		2084
2023		2085
2024	where $I_0(x, y, t)$ is the underlying noiseless video. This noise causes the complex steerable pyramid coefficients to be noisy, which causes	2086
2025		2087
2026	the local phase to be noisy. We show that the local phase at a point has an approximate Gaussian distribution when the amplitude is high	2088
2027		2089
2028	and is approximately uniformly distributed when the amplitude is low.	2090
2029		2091
2030	The transformed representation has response	2092
2031		2093
2032		2094
2033	$g_{r,\theta} * I_0 + g_{r,\theta} * I_n \text{ and } h_{r,\theta} * I_0 + h_{r,\theta} * I_n.$	[13] 2095
2034		2096
2035		2097
2036	The first term in each expression is the noiseless filter response, which we denote $S_{0,r,\theta} = g_{r,\theta} * I_0$ for the real part and $T_{0,r,\theta} = h_{r,\theta} * I_0$	2098
2037		2099
2038	for the imaginary part. The second term in each expression is filtered noise, which we denote as $S_{n,r,\theta}$ and $T_{n,r,\theta}$. At a single point,	2100
2039		2101
2040	$S_{n,r,\theta}$ and $T_{n,r,\theta}$ are Gaussian random variables with covariance matrix equal to	2102
2041		2103
2042		2104
2043	$\sigma^2 \begin{pmatrix} \sum_{x,y} g_{r,\theta}(x,y)^2 & \sum_{x,y} g_{r,\theta}(x,y)h_{r,\theta}(x,y) \\ \sum_{x,y} g_{r,\theta}(x,y)h_{r,\theta}(x,y) & \sum_{x,y} h_{r,\theta}(x,y)^2 \end{pmatrix} = \sigma^2 \sum_{x,y} g_{r,\theta}(x,y)^2 I$	[14] 2105
2044		2106
2045		2107
2046	where I is the identity matrix and equality follows from the fact that $g_{r,\theta}$ and $h_{r,\theta}$ are quadrature pairs.	2108

We suppress the indices r, θ in this section for readability. From Eq. 11, the noiseless and noisy phase are given by

$$\phi_0 = \tan^{-1}(T_0/S_0) \text{ and } \phi = \tan^{-1}((T_0 + T_n)/(S_0 + S_n)). \quad [15]$$

Their difference linearized around (S_0, T_0) is

$$\tan^{-1}\left(\frac{T_0 + T_n}{S_0 + S_n}\right) - \tan^{-1}\left(\frac{T_0}{S_0}\right) = \frac{S_n S_0 - T_n T_0}{A_0^2} + O\left(\frac{S_n^2, S_n T_n, T_n^2}{A_0^4}\right). \quad [16]$$

The terms S_n^2 and T_n^2 are expected to be equal to their variance $\sigma^2 \sum g_{r,\theta}(x, y)^2$. Therefore, if $A_0^2 \gg \sigma^2 \sum g_{r,\theta}(x, y)^2$, higher order terms are negligible. In this case, we see that the phase is approximately a linear combination of Gaussian random variables and is therefore Gaussian. This is illustrated empirically by local phase histograms of the green and blue points in Extended Data Fig. S3a-e.

For these high amplitude points, we compute the variance of the phase of a coefficient:

$$E\left[\left(\tan^{-1}\left(\frac{T_0 + T_n}{S_0 + S_n}\right) - \tan^{-1}\left(\frac{T_0}{S_0}\right)\right)^2\right] \quad [17]$$

$$\approx E\left[\left(\frac{T_0 S_n - S_0 T_n}{A_0^2}\right)^2\right] \quad [18]$$

$$= E\left[\frac{T_0^2 S_n^2 - 2T_0 S_0 S_n T_n + S_0^2 T_n^2}{A_0^4}\right] \quad [19]$$

$$= \frac{\sigma^2 \sum g_{r,\theta}^2 (T_0^2 + S_0^2)}{A_0^4} \quad [20]$$

$$= \frac{\sigma^2 \sum g_{r,\theta}^2}{A_0^2}. \quad [21]$$

The first approximation follows from the linearization of Eq. 16.

When the amplitude is low compared to the noise level ($A_0^2 \ll \sigma^2 \sum g_{r,\theta}(x, y)^2$), the linearization of Eq. 16 is not accurate. In this case, $S_0 \approx 0$ and $T_0 \approx 0$ and phase is given by

$$\tan^{-1}\left(\frac{T_n}{S_n}\right). \quad [22]$$

T_n and S_n are uncorrelated Gaussian random variables with equal variance, which means that the phase is a uniformly random number. The phase at such points contains no information and intuitively corresponds to places where there is no image content in a given pyramid level (Extended Data Fig. S3e, red point).

5. Noise Model and Creating Synthetic Video

We adopt a signal-dependent noise model, in which each pixel is contaminated with spatially independent Gaussian noise with variance $f(I)$ where I is the pixel's mean intensity (27)(36). Liu et al. (27) refer to this function f as a *noise level function* and we do the same. This reflects that sensor noise is well-modeled by the sum of zero-mean Gaussian noise sources, some of which have variances that depend on intensity (5). We show that this noise model is an improvement over a constant variance noise model in Extended Data Fig. S9.

The noise level function f is estimated from temporal variations in the input video, with observed intensities $I(x, y, t)$. Assuming that I is the sum of noiseless intensity I_0 and a zero-mean Gaussian noise term I_n with variance $f(I_0)$, the temporal variations are given by

2233 the following Taylor expansion 2295

2234 2296

2235 2297

2236 $I(x, y, t) = I_0(x, y, t) + I_n(x, y, t)$ [23] 2298

2237 2299

2238 $= I_0(x - u(x, y, t), y - v(x, y, t), 0) + I_n(x, y, t)$ [24] 2300

2239 2301

2240 $\approx I_0(x, y, 0) - \frac{\partial I_0}{\partial x} u(x, y, t) - \frac{\partial I_0}{\partial y} v(x, y, t) + I_n(x, y, t).$ [25] 2302

2241 2303

2242 2304

2243 [26] 2305

2244 2306

2245 2307

2246 The second equality is the brightness constancy assumption of optical flow (37, 38). We exclude pixels where the the spatial gradient 2308

2247 $(\frac{\partial I_0}{\partial x}, \frac{\partial I_0}{\partial y})$ has high magnitude from our analysis. At the remaining pixel, temporal variations in I are mostly due to noise 2309

2248 2310

2249 2311

2250 $I(x, y, t) \approx I_0(x, y, 0) + I_n(x, y, t).$ [27] 2312

2251 2313

2252 2314

2253 At these pixels, we take the temporal variance and mean of I , which in expectation are $f(I_0)$ and I_0 respectively. To increase robustness, 2315

2254 2316

2255 we divide the intensity range into 64 equally sized bins. For each bin, we take all those pixels with mean inside that bin and take the 2317

2256 2318

2257 mean of the corresponding temporal variances of I to estimate the noise level function f . 2319

2258 2320

2259 2321

2260 With f in hand, we can take frames from existing videos and use them to create simulated videos with realistic noise, but with known, 2322

2261 2323

2262 zero motion. In Extended Data Fig. S6a, we take a frame $I_0(x, y, 0)$ from a video of the metamaterial, filmed with a Phantom V-10, and 2324

2263 2325

2264 add noise to it via the equation 2326

2265 2327

2266 $I_S(x, y, t) = I_0(x, y, 0) + I_n(x, y, t) \sqrt{f(I_0(x, y, 0))},$ [28] 2328

2267 2329

2268 2330

2269 where I_n now is Gaussian noise with unit variance. We motion magnify the resulting video 600 times in a 20Hz band centered at 50Hz to 2331

2270 2332

2271 show that motion magnified noise can cause spurious motions (Extended Data Fig. S6b). 2333

2272 2334

2273 We use the same simulation to create synthetic videos with which to estimate the covariance matrix of the motion vectors. 2335

2274 2336

2275 We quantify the noise in the motion vectors by estimating their covariance matrices $\Sigma_{\mathbf{V}}(x, y)$. These matrices reflect variations in the 2337

2276 2338

2277 motion caused by noise. It is not usually possible to directly estimate them from the input video because both motions and noise vary 2339

2278 2340

2279 across frames and the true motions are unknown. Therefore, we create a noisy, synthetic video $I_S(x, y, t)$ with known zero true motion 2341

2280 2342

2281 (Eq. 28, Extended Data Fig. S7a-c). 2343

2282 2344

2283 2345

2284 We estimate the motions in I_S (Extended Data Fig. S7d) using our technique with spatial smoothing, but without temporal filtering, 2346

2285 2347

2286 which we handle in a later step. This results in a set of 2D motion vectors $\mathbf{V}_S(x, y, t)$, in which all temporal variations in \mathbf{V}_S are due to 2348

2287 2349

2288 noise. The sample covariance matrix over the time dimension is 2350

2289 2351

2290 2352

2291 $\Sigma_{\mathbf{V}} = \frac{1}{N-1} \sum_t (\mathbf{V}_S(x, y, t) - \bar{\mathbf{V}}_S(x, y)) (\mathbf{V}_S(x, y, t) - \bar{\mathbf{V}}_S(x, y))^T$ [29] 2353

2292 2354

2293 2355

2294 where $\bar{\mathbf{V}}_S(x, y)$ is the mean over t of the motion vectors. $\Sigma_{\mathbf{V}}$ is a 2×2 symmetric matrix, defined at every pixel, with only three unique 2356

2357 components. In Extended Data Fig. S7e, we show these components, the variances of the horizontal and vertical components of the motion 2419
 2358 and their covariance. 2420
 2359 2421
 2360

2361 The motion \mathbf{V} projected onto a direction vector $\mathbf{d}_\theta := (\cos(\theta), \sin(\theta))$ is $\mathbf{V} \cdot \mathbf{d}_\theta$ and has variance $\sigma_V^2(\theta) = \mathbf{d}_\theta^T \Sigma_V \mathbf{d}_\theta$. Of particular 2423
 2362 interest is the direction θ of least variance that minimizes $\sigma_V^2(\theta)$. In the case of an edge in the image, the direction of least variance is 2424
 2363 usually normal to the edge. 2425
 2364 2426
 2365 2427
 2366 2428
 2367 2429

2368 6. Analytic Justification of Noise Analysis 2430

2369 2431
 2370 2432
 2371 We analyze only the case when the amplitudes at a pixel in all subbands are large ($A_{r,\theta} \gg \sigma^2 \sum g^2$) because the local phases have a 2433
 2372 Gaussian distribution in this case. Such points intuitively correspond to places where there is image content in at least two directions. 2434
 2373 2435
 2374 2436
 2375 In this case, we show that the sample covariance matrix computed using a simulated video with *no* motions is accurate for videos with 2437
 2376 sub-pixel *small* motions. 2438
 2377 2439
 2378 2440
 2379 2441

2380 We reproduce the linearization of phase constancy equation (Eq. 9) with noise terms added to the phase variations (n_t) and phase 2442
 2381 gradient (n_x, n_y): 2443
 2382 2444
 2383 2445

$$2384 \Delta\phi_{r,\theta} + n_t = (u, v) \cdot \left(\frac{\partial\phi_{r,\theta}}{\partial x} + n_x, \frac{\partial\phi_{r,\theta}}{\partial y} + n_y \right). \quad [30] \quad 2446$$

2385 2447
 2386 2448
 2387 The total noise term in this equation is $n_t + un_x + vn_y$. The noise terms n_t, n_x and n_y are of the same order of magnitude. Since u and 2449
 2388 v are much less than 1px, the predominant source of noise is from n_t and the effects of n_x and n_y are negligible and we can ignore them, 2450
 2389 2451
 2390 2452
 2391 allowing us to write the noisy version of the equation as 2453
 2392 2454
 2393 2455

$$2394 \Delta\phi_{r,\theta} + n_t = (u, v) \cdot \left(\frac{\partial\phi_{r,\theta}}{\partial x}, \frac{\partial\phi_{r,\theta}}{\partial y} \right). \quad [31] \quad 2456$$

2395 2457
 2396 2458
 2397 The motion estimate \mathbf{V} is the solution to a weighted least squares problem, $\mathbf{V} = (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W} \mathbf{Y}$ (Eq. 3). To simplify notation, 2459
 2398 let $\mathbf{B} = (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W}$, the parts of the equation that don't depend on time. Then, the flow estimate is 2460
 2399 2461
 2400 2462
 2401 2463

$$2402 \mathbf{V} = \mathbf{B} \mathbf{Y}. \quad [32] \quad 2464$$

2403 2465
 2404 2466
 2405 where the elements of \mathbf{Y} are the local phase variations over time. \mathbf{X} and \mathbf{W} contains the spatial gradients of phase and amplitude 2467
 2406 respectively. We have demonstrated that \mathbf{X} is close to noiseless (Eq. 31) and our assumption about the amplitudes being large means \mathbf{W} 2468
 2407 is also approximately noiseless, which means that \mathbf{B} is noiseless. 2469
 2408 2470
 2409 2471
 2410 2472

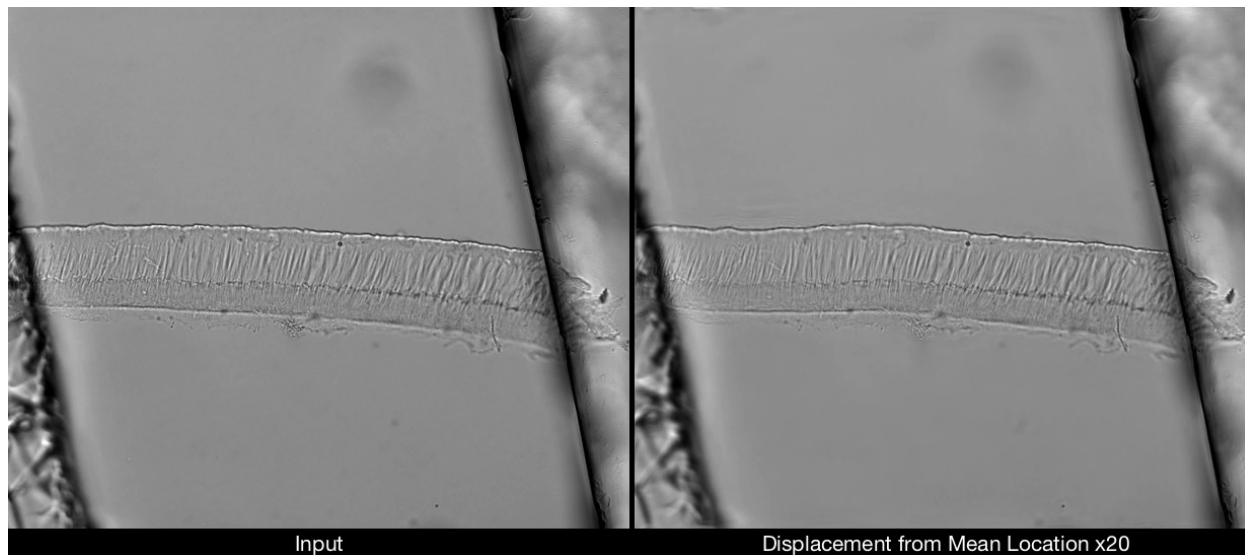
2411 We split \mathbf{Y} into the sum of its mean \mathbf{Y}_0 and variance, a multivariate Gaussian random variable, denoted as \mathbf{Y}_n , that has zero-mean 2473
 2412 and variance that depends only on image noise and local image content. Then, the flow estimate is 2474
 2413 2475
 2414 2476
 2415 2477

$$2416 \mathbf{V} = \underbrace{\mathbf{B} \mathbf{Y}_0}_{\text{True flow}} + \underbrace{\mathbf{B} \mathbf{Y}_n}_{\text{Noise Term (Covariance Matrix)}} \quad [33] \quad 2478$$

2481	The noise term doesn't depend on the value of the true flow \mathbf{BY}_0 . Therefore, the estimated covariance matrix is valid even when the	2543
2482		2544
2483	motions are non-zero, but small.	2545
2484		2546
2485		2547
2486	Movie Legends.	2548
2487		2549
2488		2550
2489		2551
2490		2552
2491		2553
2492		2554
2493		2555
2494		2556
2495		2557
2496		2558
2497		2559
2498		2560
2499		2561
2500		2562
2501		2563
2502		2564
2503		2565
2504		2566
2505		2567
2506		2568
2507		2569
2508		2570
2509		2571
2510		2572
2511		2573
2512		2574
2513		2575
2514		2576
2515		2577
2516		2578
2517		2579
2518		2580
2519		2581
2520		2582
2521		2583
2522		2584
2523		2585
2524		2586
2525		2587
2526		2588
2527		2589
2528		2590
2529		2591
2530		2592
2531		2593
2532		2594
2533		2595
2534		2596
2535		2597
2536		2598
2537		2599
2538		2600
2539		2601
2540		2602
2541		2603
2542		2604

DRAFT

2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2660
2661
2662
2663
2664
2665
2666



Movie S1. Traveling waves of the tectorial membrane revealed. The displacement from mean location of the membrane in the input video on the left was amplified by twenty times to produce the motion magnified video shown on the right. The original video consists of eight frames. The included video repeats these eight frames ten times for 80 frames and plays the result at 10 frames per second.



Movie S2. The input bridge video is concatenated with two motion magnified videos revealing different modal shapes of the bridge. Motions within a 1.6-1.8 Hz frequency band are amplified 400 times to produce the video on the right, in which the first bending mode is revealed. Motions within a 2.4-2.7 Hz frequency band are amplified 250 times to produce the video on the bottom, in which the first torsional mode is revealed. The impact of the central span (not shown in the video) occurs approximately five seconds after the video's start.

2667
2668
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2690
2691
2692
2693
2694
2695
2696
2697
2698
2699
2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728

2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749
2750
2751
2752
2753
2754
2755
2756
2757
2758
2759
2760
2761
2762
2763
2764
2765
2766
2767
2768
2769
2770
2771
2772
2773
2774
2775
2776
2777
2778
2779
2780
2781
2782
2783
2784
2785
2786
2787
2788
2789
2790



Movie S3. The input pipe video concatenated with five motion magnified videos revealing different modal shapes of the pipe. The sub-videos were scaled up using bicubic interpolation by a factor of 50%. The video was recorded at 24,096 FPS and played back at 30 FPS.

DR

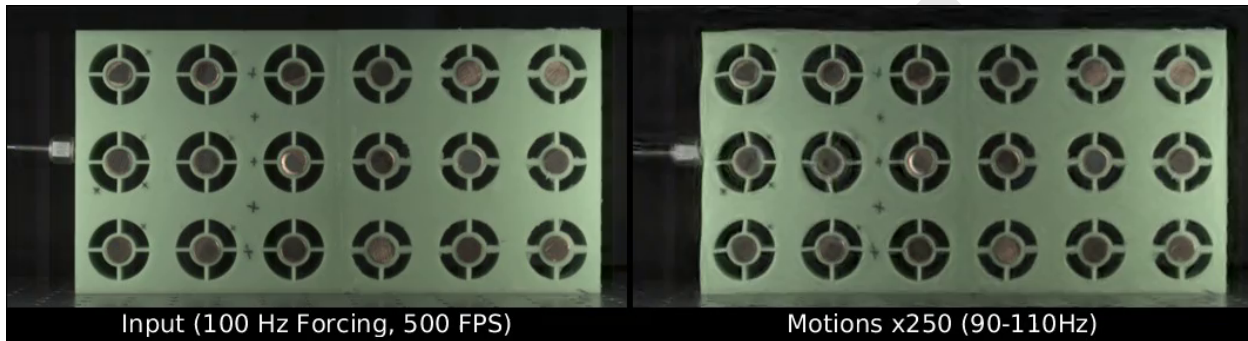


Movie S4. A probe vibrates the metamaterial at 50Hz. The input high speed video is shown on the left. Motions within a 40-60 Hz frequency band are amplified 80 times to produce the video on the right, in which the propagation of the vibrations is revealed. The video was recorded at 500 FPS and is played back at 30 FPS.

2791
2792
2793
2794
2795
2796
2797
2798
2799
2800
2801
2802
2803
2804
2805
2806
2807
2808
2809
2810
2811
2812
2813
2814
2815
2816
2817
2818
2819
2820
2821
2822
2823
2824
2825
2826
2827
2828
2829
2830
2831
2832
2833
2834
2835
2836
2837
2838
2839
2840
2841
2842
2843
2844
2845
2846
2847
2848
2849
2850
2851
2852

2853
2854
2855
2856
2857
2858
2859
2860
2861
2862
2863
2864
2865
2866
2867
2868
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2899
2900
2901
2902
2903
2904
2905
2906
2907
2908
2909
2910
2911
2912
2913
2914

2915
2916
2917
2918
2919
2920
2921
2922
2923
2924
2925
2926
2927
2928
2929
2930
2931
2932
2933
2934
2935
2936
2937
2938
2939
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2950
2951
2952
2953
2954
2955
2956
2957
2958
2959
2960
2961
2962
2963
2964
2965
2966
2967
2968
2969
2970
2971
2972
2973
2974
2975
2976



Movie S5. A probe vibrates the metamaterial at 100 Hz. The successful attenuation of vibrations are revealed in the motion magnified video on the right, in which motions in a frequency band of 90-110 Hz are amplified 250 Hz. The high-speed input video of the metamaterial is shown on the left. It was recorded at 500 FPS and is played back at 30 FPS.